Department of Computer Science

Technical Report

Undergraduate Dissertation: A Student Event Management Solution

Catherine Rackham

Technical Report 2004-07 **May 2004**

ISSN 1740-9497

Contact Address:
Department of Computer Science
University of Bath
Bath, BA2 7AY
United Kingdom
URL: http://www.cs.bath.ac.uk

ISSN 1740-9497

A Student Event Management Solution

Catherine Rackham

BSc in Computing Information Systems

May 2004

A Student Event Management Solution

Submitted by Catherine Rackham

DECLARATION

This dissertation is submitted to the University of Bath in accordance with the requirements of the degree of Batchelor of Science in the Department of Computer Science. No portion of the work in this dissertation has been submitted in support of an application for any other degree or qualification of this or any other university or institution of learning. Except where specifically acknowledged, it is the work of the author.

Catherine Rackham
May 2004

Abstract

Event management is a profession that requires a wide variety of skills, experience and contacts to be successful in, yet many students face the challenge of running an event as newcomers to the industry. They co-ordinate both small and large-scale events whilst at University with few or none of the proficiencies mentioned that results in a varying success rate.

With a high demand for assistance in this area, the project attempts to provide a facility through the framework of an Electronic Performance Support System (Gery (1991)) to give the students all the help they need in a timely manner.

The requirements and design lay the foundations for a functional prototype that is scrutinised by several potential users. While it is recognised that the resulting application is not 100% suitable for use, it goes a long way in the right direction and provides a solid foundation of research on which further development work can be based.

Acknowledgements

Grateful thanks go to Marina De Vos for all her support and inspiration throughout this project. Thanks also go to Jo Hyde and Charlie Slack for their input.

On the sanity side of things, thanks to all my trouble shooters, readers, cooks and all my family and friends who have supported me throughout my time at University. You are all wonderful.

Table of Contents

Index of Figure

Index of Tables

1 Introduction

University of Bath students currently co-ordinate many small and large-scale events, often without prior experience of having running them. Event management is something that cannot be taught, but how can they best be steered into running events in an effective, safety conscious way? Hence what can be delivered to assist this process?

1.1 Background

Students currently organise events both on and off campus: mainly for social activities, sporting events and to raise money. They receive very little training, and although organising and running an event is not rocket science, effective event managers build on common sense and experience to improve their skills.

Neither common sense nor experience are subjects that can be taught but knowledge gained is often lost after the event and not passed on to others. Simple questions such as "How do we book a room on campus?" and "What is the capacity of the University Hall?" are asked time and time again by students, yet a simple facility could prevent such time wastage.

The Students' Union runs basic general training for their society committees at the beginning of each academic year in order to give students a start, but the training is only general and very limited in scope. Several members of staff have it in their remit to help students organise events where appropriate to deliver guidance but they cannot always be on-hand?

1.2 Problem Description

Students help run extra curricular activities for fun or for the good of the group they are associated with. The university lays down many procedures constraining student activities to ensure the welfare of all involved. We need to equip students with the knowledge they need to organise and plan successful events within these guidelines whilst encouraging them and helping ensure that it is a rewarding and enjoyable experience.

So what motivates students to put down their books and take up other supplementary activities? How do they go about these tasks – in groups or on their own and hence what do they learn from such pursuits? Furthermore, what thought process does the untrained person go through when trying to discover such information - Do they feel their way tentatively? Or approach each challenge with confidence? This does not necessarily matter here, what matter is, that no student is sure of what they are doing and therefore needs to learn as they go through the various stages.

They may be other influential factors involved: Running an event can be a stressful experience – what impact does this have on an individuals performance and thus how can we reassure and manage any difficulties through an encouraging system.

Initial research into existing event management products has shown a gap in the market for this type of application. There are many pieces of software for delegate handling and for knowledge storage, but none that teach students how to plan effectively whilst guiding them through the event set-up process. One reason for this is that it is a highly specialised area. Who else would care about how you would book a microphone in Bath University? This is what we want to teach and supply knowledge for.

There is information out there for students produced by universities which are simply guidance notes – in the case of Nottingham University, a 16 page pdf document with a "fill in the blanks" type approach. Each university approaches things differently, but what is

common is that they aim to teach students and supply actual facts, rather than simply providing a general structured storage medium which commercial software does.

The aim of this project is to provide an interactive guidance package to help students manage their events within set constraints.

The project will be analysed during both the design and implementation stages and tested by real users. During the design phase, mock-ups can be produced so that users can trace through the system observing its qualities and downfalls, whilst testing after implementation will allow for them to really get to grips with the system.
Both the way they in which they use the system, and their feedback and reactions to it will be closely monitored. Any unexpected behaviour or frustrations with the system will be noted specifically, and efforts will be made to make the final system more appealing.

The ultimate aim of the project is for it to be put into use by students in the university in order for them to manage their events effectively. It must capture their interest whilst not being too simple to be ignored or too complicated to learn quickly?. Students inevitably will look for a way in which to make their job easier and this will be the perfect support tool for them.

1.3 Document Structure

This project seeks to demonstrate to the reader the challenges and interest of this project and how a solution was derived and evaluated. This includes a detailed review of the relevant research, how the system requirements and design were constructed, and whether or not the solution was a success. This dissertation has been divided into several sections in order to achieve this as outlined below:

Literature Review: The literature review will seek to investigate two main areas. Of how the system should encourage its students to organise events in the best possible way, and how the application should present itself. Currently available applications will also be analysed here and any lessons learnt remembered.

Current Student Event Management Advice: This chapter will look into the advice that is currently available to students looking to organise events at other universities.

Requirements: This requirements section shall present what the requirements for this application will be, and how and where they came from. These will form a solid basis from which the application can be designed.

Design: This section outlines how the application will be implemented and what it will look like by following established guidelines and design patterns. The design section also shows how the proposed system is validated with the users and any resulting alterations that will be required.

Implementation: Based on the design, the implementation section will show how the design was put together. Included in this will be a discussion on any differences between the designed system and the implemented one and the reasons behind that.

Testing: The application will be scrutinised here to establish whether or not it worked properly and met its objectives. The tests and their results will be presented.

Conclusion: This final section will seek to determine the project's successes and failures, and outline any areas of recommended further work.

2 Literature Review

2.1 Introduction

The proposition of designing and implementing an event management system to support students in this undertaking poses several challenges. Most students will come to University will no prior experience of organising events, yet of those that get involved, many will continue to organise further ones.

The first priority is to establish what tasks are associated with running an event, and hence to investigate what students need to know – in both the order of the planning process, and what skills they require to complete it.

Secondly, even if we knew what we had to teach students and what the relevant information was that they needed, we must still establish a way to portray this information. Furthermore, how can we teach them about event management in a way such they can learn and be able to perform?

This review seeks to gain an insight into these issues with the aim of answering them, to go forward to create a suitable application for students. Existing software is also scrutinised here to see how other programs approach the issues and what lessons can be learnt from them.

2.2 Event Management

Event management is a creative process that requires pulling a very large number of skills and features together to be successful. Every day event managers form and build up events and everything that goes around it. No assumptions can be made and no stone left unturned.
The example of booking a venue is a simple way that the complexity of event management can be demonstrated, as there is more to it than merely booking a room. Associated with it is catering, seating arrangements, audiovisual requirements, lighting, sound and of course any other needs of the guests.

This definition of event management may sound like a mouthful but that is indeed what makes it a profession.

> Event management is the process by which an event is planned, prepared, and produced. As with any other form of management, it encompasses the assessment, definition, acquisition, allocation, direction, control, and analysis of time, finances, people, products, services, and other resources to achieve objectives. [1]

2.3 Knowledge

It is difficult to know where the issues surrounding knowledge fit into this literature review. It is evident throughout each section because everything we do is based around knowledge: What we have, what we need to complete the task, and then how we use it.
This is summed up by Bush (cited by Rosenberg (2001)) who identifies four stages of using knowledge effectively:

- Collect – learn from what's there already
- Relate – consult with peers / boss throughout
- Create – explore, compose, evaluate possible solutions
- Donate – evaluate and put more knowledge back in to the system

[1] *Event Management Body of Knowledge Project* [Online] Available at http://www.juliasilvers.com/embok.htm (accessed November 10 2003)

Aberg and Shahmehri (2001) say that “An important challenge for universal usability is to bridge the gap between what users know and what they need to know to successfully interact with a computer system”.
This can be applied to the concept of any task, and especially to managing an event. Users need just enough information to complete a task without being overloaded, and this is discussed further in the section entitled “Why Aren’t Electronic Performance Support Systems as Developed in the Corporate Setting?”

2.4 Planning and organising – is there one best way?

In order to design a system to lead students through organising an event in the best way, it is necessary to now look into how to do that to be able to structure the system appropriately. This section looks at some of the established methods behind project and event management, around which the high level system design can be built. As this project aims to provide an overall tool for students organising events, it is also appropriate here to look into some of the skills required to be a good project/event manager and section 2.5 focuses on some of the key issues.

Event management is a specialised form of project management and a selection of key methods will be compared here to establish an optimum way to encourage students to manage their events.

2.4.1 The project management approach

There is a lot of research available in the area of project management, and many models constructed by experts conveying how projects can be run. In reality, every project is going to be unique, but these approaches provide a framework around which projects can be based.

One of the most widely used approaches to project management, which is also particularly relevant to IT projects, is the PRINCE2 method. It identifies 8 key stages of project management, which are summarised in Table 2.1.

	Stage	**Description**
1	Directing the project	Done by those in charge of the project, e.g. senior management
2	Planning the project	Ongoing
3	Start up	Organisation resources are identified
4	Initiating a project	An overall project strategy is defined– how it will be judged
5	Controlling a stage	Control, problem solving
6	Managing product delivery	Vital for multi-team projects
7	Managing stage boundaries	Reporting, progress
8	Closing project	Shutdown, reports, feedback

Table 2.1 Summary of PRINCE2

Each of these stages is very useful, but it is geared towards a standard project, that runs for a set duration and covers a specified number of ordered tasks that lead to its completion. The difference with an event is that there are almost two projects – planning it, and later running it. Planning typically required 99% of the time to complete the event project whilst managing delivery of stages, and controlling the project – including problem solving are typically involved in the event itself - during setup and then the execution.

Deming (1996) presents a cycle for constant improvement. – as cited by Maylor (2003) shown in Figure 2.1.

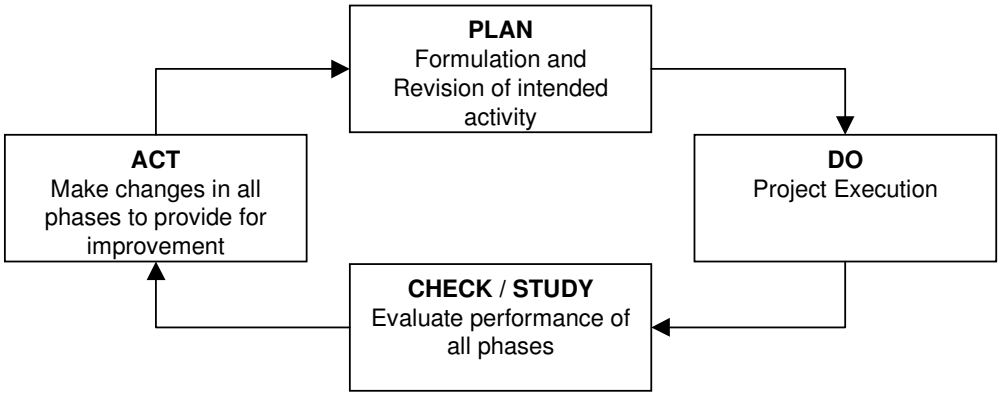


Figure 2.1 Cycle for constant improvement

When applied to the area of project management, the model above translates into the concept of 3 D's:

Stage	Phases
Design it	Concept, analysis, proposal, justification, agreement
Do it	Start up, execution, completion, hand over
Develop it	Review, feedback

Table 2.2 Three D's model

This model re-emphasises the significance of evaluation in the project management process, in line with what is consistently mentioned in Part 1. It encourages people to learn from their mistakes and to use them as a basis for improvement. However, as discussed in section 2.3, the forth stage of using knowledge effectively is to donate knowledge back into a system to allow *others* to benefits from the mistakes you have made, and the things you have learnt.

Where the concept of planning has been mentioned in the models above, in event management, planning is especially important, because of the large amount time spent planning an event, and the relatively short time spent actually running it. The life cycle of planning delves into this stage a little further.

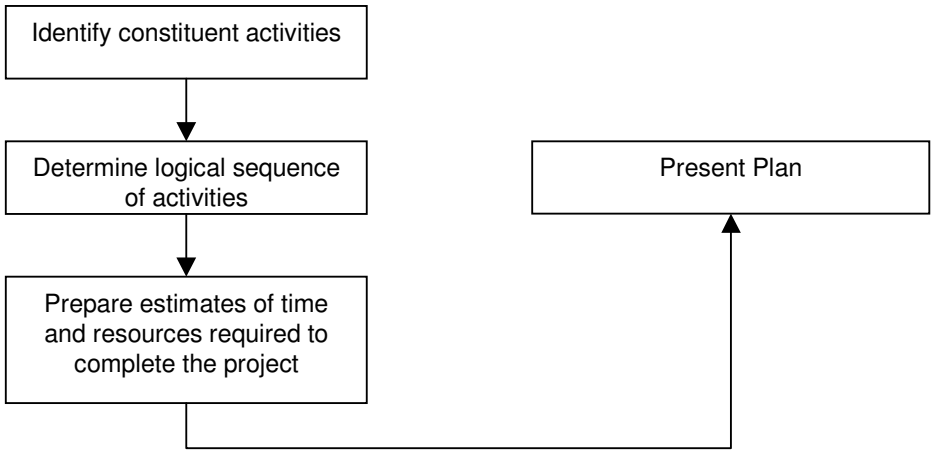


Figure 2.2 Planning Life Cycle

One thing that has not yet been considered is whether a system should tell students not only what to do, but what order they should do things in. This is a flexibility that could be given to them, to enable them to make decisions for themselves, but the system can be used to support this, make suggestions and highlight time critical stages where necessary. For example, if a world record was being attempted, it takes at least 4 weeks to simply get hold of an information pack.

2.4.2 The Event Management Approach

Similar to the field of project management, there are also several approaches documented on how to prepare for, plan and run an event.

Torkildsen (1999) presents 7 distinct stages summarised as follows:

1) Idea
    a. Formulate Ideas
    b. Receive proposals
2) Feasibility
    a. Evaluate proposal
    b. Consider problems
    c. If the event is *not* feasible , say so
    d. If the event is feasible, set aims and announce decision making progress
3) Planning Strategy
    a. Appoint organising committee
    b. Appoint co-ordinator
    c. Set objectives, structure, budgets, targets, dates and strategies
4) Structure Units / Teams / Responsibilities
    a. Programme, staff, administration, finance, marketing, technical, catering and services
5) Detailed Preparation
    a. Flow chart, monitoring, checklist, meetings, staff training, practice and double check
6) Event
    a. Good start
    b. Good flow
    c. Good finish
7) Evaluation
    a. Feedback
    b. Evaluation
    c. Modification for next event

This is a formal, but very comprehensive method of organising an event.

Watt (1998) presents a similar approach, as seen in Figure 2.3. Like Torkildsen's (1999) methodology, it begins with idea and feasibility studies, ends with the event and evaluation and describes the planning activities in between. In addition, Watt's model identifies the reason why events are usually run – for a customer. Their needs and objectives are key to the whole process because meeting customer needs indicates in part the success of the event.

Watt also mentions that monitoring is not just useful during event planning, but also during the event itself. This is backed up with examples by Wagen (2001) and Shone (2001) as discussed previously in section 1.4.

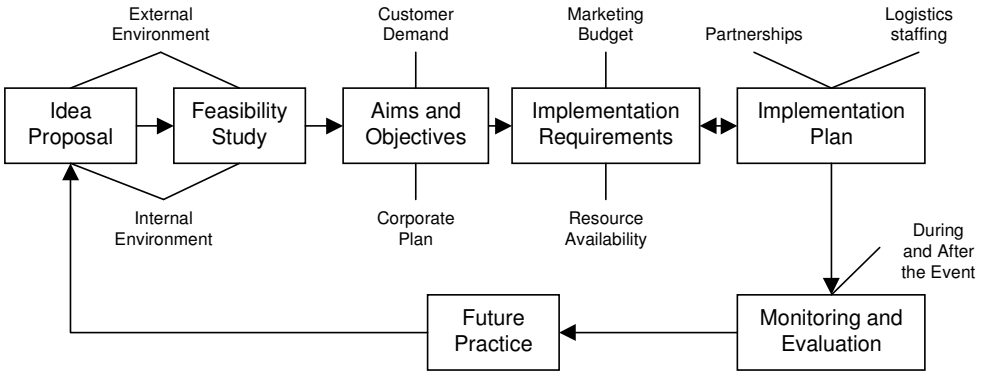


Figure 2.3 Watt

One concern about these models is the rigidity of the stages they portray. Figure 2.4 shows the planning structure that was followed by the Students' Union whilst organising Freshers' Week 2003. Due to the time constraints, it was necessary for volunteers to be recruited before any event ideas were put together, and any decisions made on what events to run. Whether this is an exception to the norm or a frequent occurrence is irrelevant, but a system that is to be used for all different types of events must not constrain organisers with its structure.

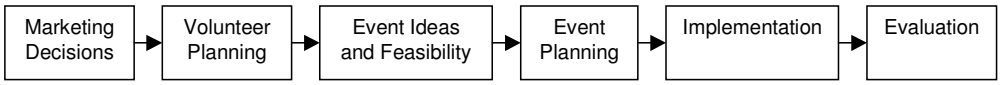


Figure 2.4 Bath University Students' Union Freshers Week Planning Structure

To summarise, all of the project management and event management methodologies cover four main principles:

1) Objective setting (Idea, feasibility)
2) Planning (detailed planning, control)
3) Running the Event
4) Evaluation and closure

Although such an order is very useful to base a design of the system on, it may not be possible for an event manager to follow these stages in all circumstances. Therefore, the system needs to be flexible, and designed such that it will not restrict its users but will support and guide them, whilst instilling some thought into their planning.

The only mandatory stage for event organisers dictated by the Students' Union is to complete the "£200+ event form"; It must be filled in by the event team before they can get the go ahead to run their event. As it is largely budget based, quotes need to be gathered in order to estimate costs reasonably, so preliminary planning will be done whilst the feasibility study is taking place. This in itself, is another demand for flexibility in the planning structure of the system. A copy of the "£200+ event form" can be found in Appendix A.

2.5 Related Project Management Skills

Just as no human will ever be perfect, no project manager will handle every situation in a perfect way. Management researchers will argue that there is no one best way of managing, as it depends entirely on the situation, the context, and the people involved. However, there are skills that can be taught and a system which teaches students to manage events must aim to make students aware of these topics, and teach them about

them in a relevant way to their needs in order for them to learn. Being able to do these things well should only have a positive impact on the way they organise their events.

2.5.1 Scheduling

The larger the project and the more people involved in it, the greater the need for an effective planning schedule.

Projects are often divided into many tasks, which are then split into sub-tasks and so on, which are usually inter-linked with various dependencies. For example, you would not want to confirm the number of meals required for your staff, until you knew how many staff you had.

In the student event management context, scheduling is difficult as we do not want to dictate to students what planning they should be doing and when. Not only are they are volunteers, but have varying pressures of work and other commitments – especially due to the varying week by week demands of being a student. A schedule is likely to simply not be adhered to kept to and hence become meaningless.

There are many methods that can be used to arrange schedules and record them – e.g. Gantt charts, critical path analysis etc, but there are programs available should students want to use them and as discussed in section 2.10.1, an all-encompassing system is not necessarily the most useful. However, students should be made aware of their usefulness, and appropriate training on using them could be included, should they wish to use them.

2.5.2 Problem Solving

We use knowledge to help us solve problems, and this is also intrinsic to event management as decisions and problems are made occur at all stages, both whilst planning and on the night.

Shneiderman (1999) cites Couger (1996), who offers a five-phase problem solving methodology:

1. Opportunity, delineation, problem definition
2. Compiling relevant information
3. Generating ideas
4. Evaluation and prioritising ideas
5. Development and implementation plan

I do not believe that this is particularly applicable to event management, as it is very formal. The process of organising an event is not like this as there are so many decisions and things to do – which is its very uniqueness. Problem solving is often done on the spot or a very quick decision is made, so there is no time to go through the stages laid out above. However, even when a solution to an issue has to be found rapidly your mind still scans through the range of possibilities and chooses the most appropriate; this is a skill in itself.

Watt (1998) applies similar principles to event management, injecting the practical tip of “Analyse the situation as coolly as possible” and also a final stage of “Continue monitoring to avoid repetition”. This is crucial, and fits with our ethos of using evaluation to improve future events and avoid the same problems.

Having said this, I do not believe that it is always possible to run through all these formal steps due to the time available. Although good event organisers should be able to avoid as many crises as possible through good planning, staff must be ready for anything and be able to respond to any issues appropriately.

> "At most events, the pace is so fast that it is crucial that [both volunteer and full time event] staff be in a position to make decisions on the spot." [2]

Shone (2001) deems that both states are commonplace – having to act on a situation before you know all the facts, and that there are occasions when it is vital to be aware of the factors that have led to the situation before taking action.
However both Watt (1998), Wagen (2001) and Shone (2001) all agree that recording the situation and how it was solved, and then evaluating it, is crucial; both to make sure that the unfavourable situation does not happen again (as it may not be so easily solvable the next time around) and also so that it can be avoided in future events all together.

If a problem occurs whilst an event is in progress, it is very unlikely that students will have either the time or equipment available to access the system for a possible answer. However, by the way students are encouraged to plan for contingencies, have access to previous problems encountered and how they were solved; this can increase the likelihood that they will be able to handle their situation(s) effectively.

2.5.3 People management

As Meredith (2003) reminds us, the biggest part of project management and, usually, the most expensive resource, is the people. One major potential hindrance to a project is conflict between members of the workforce. Meredith identifies the 3 general categories of conflict:

- Different groups have different goals and expectations
- Uncertainty over authority
- Interpersonal conflicts between people.

By the nature of the voluntary sector, people do not often get forced to work with others they don't want to, because they *choose* to participate, hence minimising potential conflict. However, these things cannot always be predicted, and a good working relationship at the beginning of a project can be much less pleasant by the end of it.
Organisers of student events are often friends and so the personal relationships already in place help develop semi-professional understanding between them.

Event managers need to be aware of any prospective conflict between members of their team, and know how to deal with it. It is difficult to teach such skills, but educating students in how to spot such situations and giving them tips on handling them effectively, may be useful to them.

People management is only one example of a skill that a piece of software cannot teach, yet may prove very valuable to them. Other examples are Leadership, motivating, handling pressure, decisiveness, etc. We often develop these skills through experience and later evaluating the effect they had. Although it may be impossible to teach these skills through software, it could offer guidance and training to allow students to think about how to handle any potential issues in a training environment.

2.5.4 Project Completion and Handover

Section 2.4 re-iterated once more the importance of evaluation and an effective handover of information for the benefit of others.

Maylor (1999) describes 8 aspects of project completion that the project manager needs to consider during this stage:

- Ensuring that there is an incentive for the project to be finished and that activities are completed

[2] Van Der Wagen, L (2001) *Event Management: For Tourism, Cultural, Business and Sporting Events.* Elsternwick, Australia: Hospitality Press Pty Ltd.

- Ensuring documentation is provided
- Closing down the project systems, particularly accounting
- Constructing an immediate review of activities – providing a starting point for all improvement activities
- Disposal of all assets that are surplus to requirement
- Ensuring that all stakeholders are satisfied
- Providing the best basis for future projects
- Providing the basis for future reviews of activities

These points can be used to design a system that encourages students to complete a project, and make evaluating the event as easy as possible, to get the most valuable feedback from them. This information can then be put back into the system to benefit future event organisers.

The proposed system must encourage students to organise their events effectively and in a logical sequence, whilst being flexible enough to cope with the planning requirements of every event.

Project management is a unique skill developed through years of experience. Whilst we can give students help and training on how to run a project, we cannot formally define the procedures they must go through as it will vary depending on each unique situation. What can be done in the scope of this system is to provide students with ideas and training to help them better handle the situations they face.

Having looked into the practical skills and schedules surrounding project and event management this review will now look at how these can be communicated to the user.

2.6 What do we need to know about Learning?

Rosenberg (2001) defines learning as our internal way of processing knowledge.
We spend our lives learning; from the basic tasks of eating, walking and talking, to studying and learning to drive, the list is endless and so is our capacity to learn.

Rose (1999) reminds us that even though everyone has an individual learning style, there are common steps everyone needs to take to learn effectively. The brain has over 100 billion brain cells and only declines with lack of stimulation.
He goes on to describe the two broad approaches to learning.
A linear approach depicts a gradual build up of information (by both amount and complexity) in a very step-by-step manner. This is very common in schools today; learning starts with very basic tasks such writing your name or reading a few words. This then gathers speed and in several years we are able to write an entire novel and read whole books.
The second approach is where an idea or learning objective is viewed as one big picture - known as the global view. The author uses the example of meeting somebody in the street to show this: When we see a person, we do not look at their hair, then their forehead, eyes, ears nose etc all in turn; we glance at somebody and immediately recognise them.

This can be related to an event. As seen in the event management section, when building an event there are a huge number of things to consider. We often see broad areas for development, such as marketing, sponsorship and decorations. Unusually perhaps to the creative aspects of the event process, we may simultaneously have ideas for the smaller touches too – such as wanting to give out a free drink to guests on arrival.

Any application supporting events needs to consider the needs of both approaches. A global picture showing the progress of all the different planning section of the event is essential, while in some areas care must be taken to ensure that processes are built up slowly to ensure that nothing is missed.

So what about delivering knowledge online?

2.7 What is E-learning?

Rosenberg (2001) explores what the E in e-learning actually stands for.
It is branded as an "electronic" way of learning, but it encompasses far more than just this. It can revolutionise how information, and better still knowledge, is managed and hence used by a large number of people to share and consequently use it to improve how things are done. This can increase an employee's level of experience to an extent, even though they have not completed the task themselves they can still learn from the experience of others. Storing information electronically also means that it can be advanced and evolved to respond to company demands, and incorporate new and improved knowledge. It does not stand still, just like the business world around us and the changing needs for knowledge.

> "Electronic learning denotes learning environments consisting of hardware, software and personnel; a multi-faceted learning programme that utilises distance learning, interactive cable TV, and the Internet to connect learning environments to homes, places of work, and the community at large." [3]

Rosenberg (2001) divides the needs of learners into three high level areas.

- Access
- Comprehensive approach
- Balance

It may seem obvious but if there is no access or time to reach the learning environment, then quite simply the employee or student cannot be trained in what the environment provides.
What makes an event manager a professional is that their job encompasses absolutely everything. This is not an easy task so by its very nature the system needs to tell students everything they need to know and cover everything they need to do. Anything less than a 100% inclusive system makes the project worthless but this is not an easy thing to achieve. Also in this context, a point that may be overlooked is ensuring the consistency in information quality which the system references. For example, the University of Bath Staff Association publishes a set of rules and policies governing its hire to non-members. As of last week, one of the critical rules was inaccurate. These anomalies need to be highlighted to the user as far as possible to avoid confusion and in turn further increase the users trust in the system.
Finally, the system should be as self-explanatory as possible. A desired balance needs to be found between the amount of training the user can expect, and what they really need. Training should be no longer than necessary, and ideally the system should require no training to be able to use it effectively.

2.8 Computer Based Training (CBT)

Whilst we talk about e-learning in the context of this support system for students organising events, our primary objective is not for them to learn about event management it is to help them successfully achieve their end result of running an event.
A form of CBT can help them learn to do this.

CBT can present the topics surrounding a subject in a logical manner, using both theory and real life examples to help students learn. CBT is often criticised and has caused a lot of frustration to a whole host of users for different reasons.

---

[3] *Definition of E-learning* [Online] Available at http://www.educause.edu/ep/ep_subjects.asp (accessed December 1 2003)

Rosenberg (2001) identifies some of these as: Poor design, old content, inaccurate, useless after initial training, no support, boring.
Presenting too much information could easily overload a student but event management is all about getting a lot of things done simultaneously and handling multiple tasks effectively. Time is often crucial. Delays in a decision or completion of a task can have considerable consequences such as drastic price increases.
This issue of multi-tasking cannot be skipped over, in this context as it is the very soul of the subject.

Usefulness is another specific concern for this project. There is no way that everything about running an event can be covered in a form of training, but if we oversimplify the system and cannot provide any help to students then there is no point in continuing.

Furthermore, every event will have numerous issues whilst it is in progress that cannot be foreseen so it is impossible to tell students how to handle every possible situation. When it comes down to it, they will be on their own and are unlikely to be anywhere near a computer whilst the event is in progress.
We can however lay down some basic guidelines such as "do not panic" and help students practice handling problems and ad-hoc queries. Soloway (1996) talks about "edutainment" – entertaining students (in an online game for example), whilst also educating them.
Event management may now begin to sound fairly daunting which it isn't so users need to be put at ease. One way is to show them success and failure stories. The experience of others is a very good learning tool. Listing "laughable" stories (as long as the culprit doesn't mind) may humour the user, and although they may laugh it is still a valuable lesson.

To conclude on CBT, it would not be a plausible solution to helping students as it has shown that there is no way it can be all-encompassing. Events are an ongoing process that which requires a lot of time and information to complete. Therefore we need to take a side step here and look into developing a medium to provide a way that events can be organised and provide the information required in a logical way.

2.9 Electronic Performance Support Systems (EPSS)

Gery (1991) classifies an EPSS as a system that permits anyone to perform a task as an expert without any training, allowing the employee to start being productive from day one.

Rosenberg (2001) highlights the advantage of this. For a business training means that employees are not working, otherwise known as 'down time'. This is effectively costing the company money as they are not being productive and the training is likely to add further costs.
This is the same for students. The easier and quicker we can make their training for event management, or eliminate it completely and let the system train them while they do it; the less hassle it is for them, and the more time they can spend getting on with the task.

We can formally define an EPSS as:

> Applications designed to run simultaneously with other applications or embedded within other applications that provide support for the user in accomplishing specific tasks. An EPSS may provide needed information, present job aids, and deliver just-in-time, context-sensitive training on demand. [4]

[4] *WBTIC Glossary of Online Learning Terms* [Online] Available at http://www.webbasedtraining.com/primer_glossary.aspx (accessed December 1 2003)

The concept of scaffolding is discussed by Soloway (1996). Scaffolding addresses the learners' growth, diversity and motivation and refers to providing support to learners to engage in activities that are normally out of their reach.
Rob Smalldon (a professional Event Manager) once said "Event management cannot be taught - it is about experience and common sense". Students may be distinctly lacking in this kind of experience, so event management is out of their scope if it is not supported. Support is currently supplied by members of Students' Union staff or by friends with their own experience, which is why I want to provide a computer system to give everyone an opportunity to manage events with ease.

2.9.1 Consumer Electronic Performance Support Systems

There are many programs around which help us improve our performance, and two obvious examples are the word processor and the spreadsheet application. Schneiderman (1999) reminds us that just a few decades ago such high quality formatting techniques required advanced skills and machinery available to only those involved in the publishing industry. In this day and age we can produce smart high quality papers, legal briefs and other documents from our home computers swiftly and easily.
The spreadsheet, also known as "what if" software, was pioneering in its field and displays one of the fundamental theories of performance support. It completes complex and often lengthy calculations for the user to free them up to concentrate on the real problem.

Another feature of EPSS's is that they require very little or no training to use allowing the user to use them immediately. In using them you do not realise that you are being trained to use a system, you simply get on with the job in hand.
One set of examples of these are online booking systems. An implementation of this can be seen at easyjet.com where customers complete a simple five step process to secure their chosen flights. Each user is guided through the process with no effort required to understand what is going on and the system adapts to the user as the process continues. For instance, even though you have already selected what language you wish to view the site in, if you enter a British departure airport prices are shown in GBP, whereas if a different country is selected the fare is shown in the currency of that country.

There is however a negative side to these performance enhancing systems, in that it can be possible to take them a step too far. In the example of the word processor, the spell checker is a useful feature to show us where we have any potential spelling mistakes. It often highlights words which it does not have in its dictionary, but then changes others automatically that it believes it knows the answer too without needing to alert the user. On many occasions, Microsoft Word has changed words in documents incorrectly. Also as we no longer need to use our brain for spellings, we have almost become reliant on the capability of the word processor. This can lead to deterioration in our own spelling abilities, and also a decline in our confidence in it. This is no positive thing.

Gery (1997) describes to us how the difference between consumer and corporate software is extensive and still increasing. She believes that this is because of the different assumptions made about consumer and large-scale systems, which she lists in her paper. Another explanation is that large-scale systems are designed to meet a set of criteria and developers implement these requirements and generally nothing more. In contrast, consumer software vendors have to care about their product and inspire users to purchase it, whilst looking after a volatile customer base.

2.9.2 Why Aren't Electronic Performance Support Systems as Developed in the Corporate Setting?

In the past machine capacity and data transmission were very expensive and far outweighed the cost of an employee's time. Due to this expense balance early data input systems were designed solely around the data and the screen therefore had as much

data entry capacity on it as possible – before the input clerk would 'submit' it for processing.
Nowadays the cost balance is the opposite way around. The cost of data processing is now far cheaper than the human expense so you would expect that the way we design systems would have altered accordingly. Despite this system design and development are still largely focussed on data, with the task to be completed coming a weak second.

Howell (2001) cites Gery (1991) who says: "In our pursuit of solutions we have assumed that the future should be an extension of the past. We need to take a step back and re-examine the tasks at hand and if the structures in force are really the best ones for the job".
Howell (2001) goes on to say that not only do developers need to take not of the tasks that need to be done, but also what people have to do to achieve them and how the work presents itself.

Howell (2001) worries that we are designing to accommodate the machine but it is the poor users' task to get the job done.
Today's world is all about information gathering, handling and manipulation, it is easy to slip into the habits of designing software around the data available but performance centred design offers an alternative. Maximised performance comes from the confluence of giving the user just enough information at just the right time to that they may carry out the tasks at hand efficiently and effectively." This is illustrated in the diagram below by Dickleman (1996).

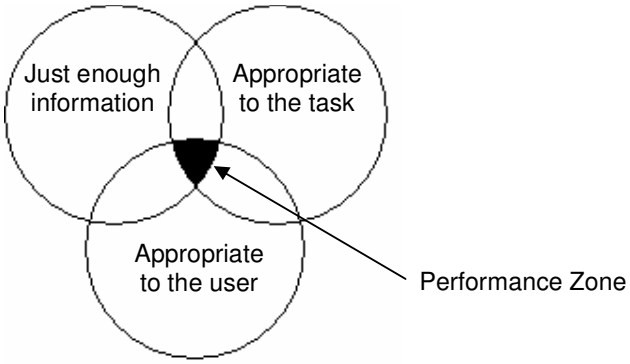


Figure 2.5 The performance zone - Dickleman (1996)

Howell (2001) goes on to cite Raybould (1995) who makes another key point to back up the above diagram – "Provide good support all about not concentrating on the components (user, information, system) but in how they work together productively."

2.10 Surely this problem has been solved before? - Existing Software

Despite an extensive investigation via an internet search engine, multiple free downloads, and numerous promising looking websites there is no software which tells students how to run events – or anyone for that matter. This is understandable. With so many possibilities and types of events, the scope of such a project would be very large to take on and there would not necessarily be such a good market for it. Each event will always have its differences and each company their favoured suppliers, which means that to capture the market software systems provide a database centred system to allow companies to manage their records themselves.

The system design required for my project is different to this. To be successful in its aim of supporting students who may only ever organise a single event, it needs to tell them

what they need to know: tell them who the contacts are that they need, suggest suppliers and outline guidelines that need to be followed.
Even though there is no existing software meeting a similar specification to what I require, there are many systems designed to support event managers, to help them manage their records effectively. I have scrutinised five such systems, and a direct comparison of their features and capabilities can be found in Appendix B.

Ideally in order to assess the real value of these systems, formal evaluation techniques would be used. Due to the cost of purchasing such a system, as well as the time constraints of having to locate users willing to be analysed, this is out of the boundaries of this review.

The systems chosen were:

- Consensus [5] – aims to support the whole process of marketing and organising events, from initial planning through to reporting after the event, and everything in between.
- GEM [6] – aims to provide a means for event managers to completely record and track all the details of any event.
- Event Pro Planner [7] – claims to be the leader in event management software, and aims to support users in their tasks within their own facility. This company have several versions of software to suit different areas, e.g. Church, Exhibition Centre. Although this could be seen to be more of a "centre management suite", it is designed to support in house event managers and provides the same kind of features to the other systems looked at here.
- Kinetic [8] - provide software for organising, booking and running events within the event manager's facility.
- Event Master [9] - has been designed and written to support the potential tasks that anyone who is involved in delegate management and organising events needs to perform.

Unfortunately there were only screenshots available for Event Pro Planner and Event Master, so most of the design criticisms centre around these two.

2.10.1 Features

The only feature that all five event management systems have, is the ability to produce reports on the various kinds of data that they hold. For example, an event plan detailing what is happening and when, and financial summaries.

Event Pro Planner and Kinetic also allow the user to create their own reports to isolate the information that they need –for example, a list of all customers who's payments are overdue by at least 3 weeks. This is a very flexible characteristic of the system – one thing to remember here is that event managers cannot physically carry their desktop computers around with them – there is currently no such thing as an event managers paperless office! However, one of the conference systems does offer a Palm Sync facility that gives it great portability.

---

[5] Welcome to Consensus [Online] Available at http://www.consensus.co.uk/event.html (accessed November 15 2003).
[6] GEM Features [Online] Available at http://www.2ls.com/index.php?page=gem (accessed November 14 2003).
[7] Event and Facilities Management Software by EventPro [Online] Available at http://www.eventpro.net/ (accessed November 19 2003).
[8] Kinetic Solutions Ltd - Conference software with a difference [Online] Available at http://www.kineticsolutions.co.uk/00000015.html (accessed November 19 2003).
[9] Mission Business Systems and Event Master [Online] Available at http://www.event-master.com/about/index.html (accessed November 15 2003).

Most of the systems also handle event finances. Budgeting, invoice tracking and billing management are integrated into the overall package to ideally decrease the amount of work required. I.e. a cost added to the budget may cause an invoice to be generated, or one to be expected.

Customers and suppliers are key – without either of them, there is usually no event, so tracking communications between them is important.
Attendee management is crucial – knowing who they are, what their movements will be and any special requirements they may have is of utmost importance. These are the people the event is being run for and who are usually paying for it. All the systems apart from Kinetic have a built in feature to support this in their software, but with such a variety of customers, and a variety of information required about them, it may be difficult to understand the system – as seen in the Design section below.
In the student context, a complex attendee management system may not always be necessary, depending on the data they require. If they are simply selling tickets on the door of the event, the only thing they need to know about their customers is how many of them there are – for capacity and fire considerations.

"Effective communication leads to good co-ordination" [10]

Knowing what suppliers are doing at all times is also crucial, as it is the event manager's job to co-ordinate when their activities occur. For example, you would not want chairs to be laid out before the carpet has been laid. Whilst planning events, there are often many negotiations between suppliers and event managers, so it is important to track what has been discussed and agreed.

If there was one thing that is crucial to all event managers by default, it is the event plan or run sheet. Wagen (2000) refers to them as indispensable. The run sheet tracks what is happening and where and when activities occur for the duration of the setup, event itself, and the event take down. Three of the systems have the capability to create event plans, and use the information in other parts of their system to enter most of the details required automatically.

Even though no checklist for an event can be exhaustive because each event will have its own unique features and circumstances, there is a core list of things which must be considered for all of them, e.g. health and safety. Event staff in Bath University Students' Union often plan events by building up checklists as items are thought of or added into the event, and then delegating tasks for completion to the relevant parties. Event Pro Planner and Kinetic both facilitate "Action Lists" within their software, but as previously explained it is impossible to comment on their usefulness here. Kinetic takes the extra step and adds some action points into this section automatically depending on what has been added into the system elsewhere, e.g. If a new supplier has been added.
This would be useful for students as they will often not know what they need to do with regard to booking a supplier for example – they will need to get a copy of their Public Liability insurance. This information would normally be stored in an event manager's brain or in company procedures which is not in existence for students in Bath.

Some of the packages, especially Event Pro Planner seem very all encompassing but I am not convinced that this is a good thing. A line needs to be drawn somewhere as to where a useful event management application ends, and another separate application begins – in the given context.
Taking the example of budgeting, Event Pro Planner has an in-built tool to assist the event manager to construct a budget. But it may be that to learn how to use this, would be more time consuming than simply advising the user, and possibly supplying a template for them to use their normal spreadsheet package – like Microsoft Excel. This may be

[10] Watt, D (1998) *Event Management in Leisure and Tourism*. New York: Adison Wesley Longman Ltd

very useful for professional event managers who use the tool every day, but it may not be so beneficial or time-efficient to a one off user for the extra training and understanding that would be required.
There will always be a cross over, as is seen in a lot of products, so the key issue here is where to draw the line in the bath student situation. One example to illustrate this point is that Microsoft Word has basic graphic manipulation built in to it, but for anything more complicated you would be expected to use a graphics program.

Other interesting and potentially useful features include a badge making and printing facility, floor plan designer and previous event information recall.
Both GEM and Event Master have a specific badge design and printing feature and whereas Event Pro Planner doesn't, it allows information to be exported to Microsoft Office products and word does have the capability of creating badges (labels). This links to the discussion in the previous paragraph as to where the line is between effective event management software and other useful standard applications.
Floor plans are an important part of the event plan as previously discussed. It would be difficult to describe to a team how to set up 200 tables and 1000 chairs using words alone, so a floor plan designer can make this as simple as possible.

Having discussed the benefits of evaluation and information sharing in section 2.3, it is good to see that three of the systems support previous information recall – allowing users to see what they did, when something was planned, and for example, what suppliers were used. Not only is this useful information if the event manager wants to use that supplier again or recommend them to someone else, but also to recommend that they should be avoided.

2.10.2 Design

With such a large amount of information to handle, there is a risk of overloading the user which could hinder their progress with a task rather than support it. GEM, Event Pro Planner and Event Master all take a modular approach to data handling and organise data into distinct sections.
Whilst this is a good idea, a characteristic of all events is how features intertwine in some way to form it. Two of EPPs section heading are "Customer" and "Accommodations" – surely customers stay in accommodation, but the link between these sections is not obvious.
It can be a good thing to set up such an obvious structure for a system as it can help the user to visualise how it all fits together in their mind. But confusion can occur when this structure is muddled. For example, on EPP Customer Screen, Figure 2.6 there is multiple use of the word "Ticklers" which can leave users bewildered as to what the difference is, if any - if they are stored in different places for example.

Figure 2.6 EPP Customer Screen

Having an all singing all dancing event management system in front of you is all very well, but Event Pro Planner allows the user to set up an event quickly and easily with a wizard. This is not only a simple way to start using the system but also ensures that no critical data is omitted here. Setting up an event according to the data that is required poses the risk that data may become the centre of a systems design, as discussed in section 2.9.2. This can be recognised in Figure 2.7. With so many buttons, tabs and symbols, it is not only difficult to understand what the information is showing the user, but also hard to distinguish what the user is to do with it all.

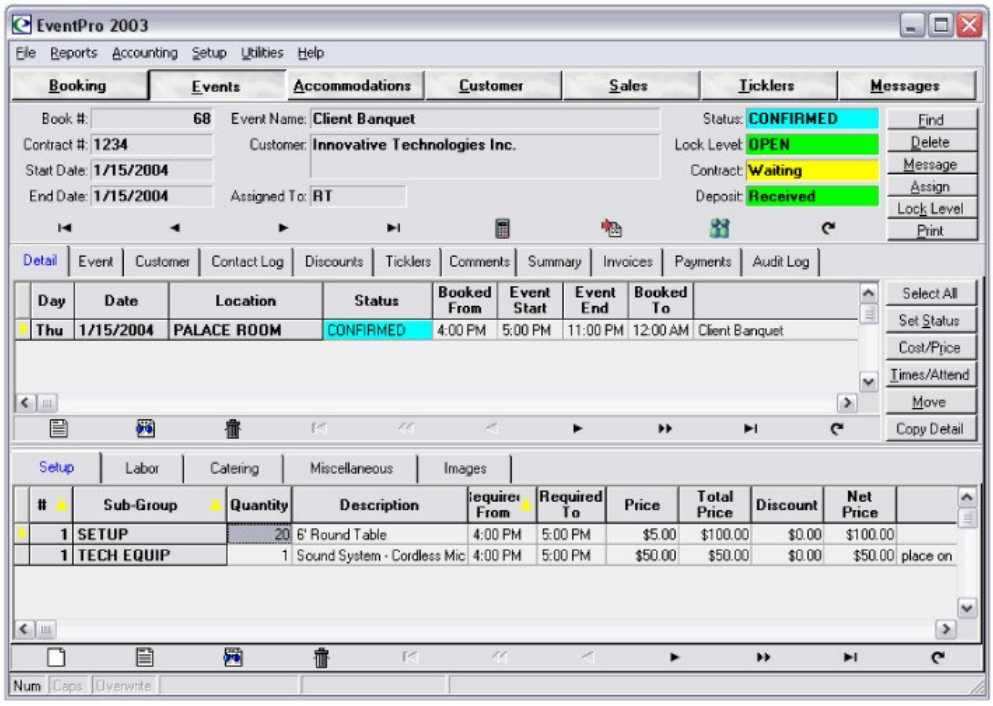


Figure 2.7 EPP Event Page screen shot

Screen clutter is another issue that overlaps here and Figure 2.8 shows the very data-centric nature of this software. Having tabs at the top and bottom of sections can be both confusing and disorientating.

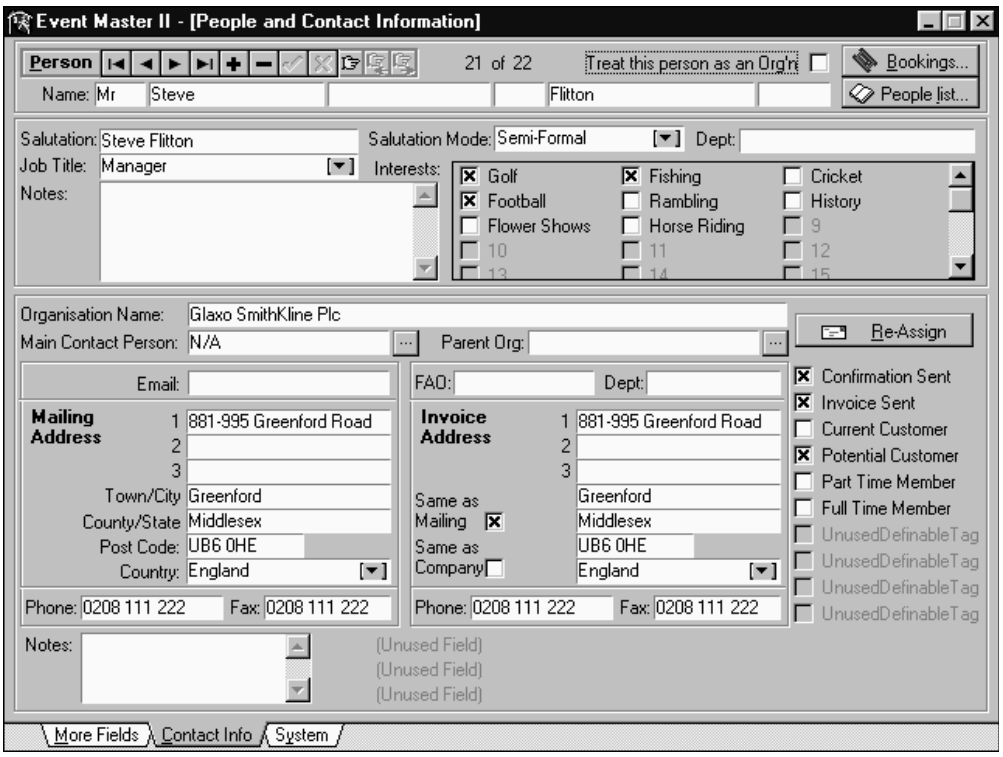


Figure 2.8 Event Masters Contact page

If a piece of software uses colour well, that compliment each other and are not overused, it can enhance a users experience of it. However systems like EPP highlight the importance of having an aesthetically pleasing presentation because a screen that is repulsive to look at does not have good potential for persuading users to work with it on a regular basis. An example of this in EPP is shown in Figure 2.9.

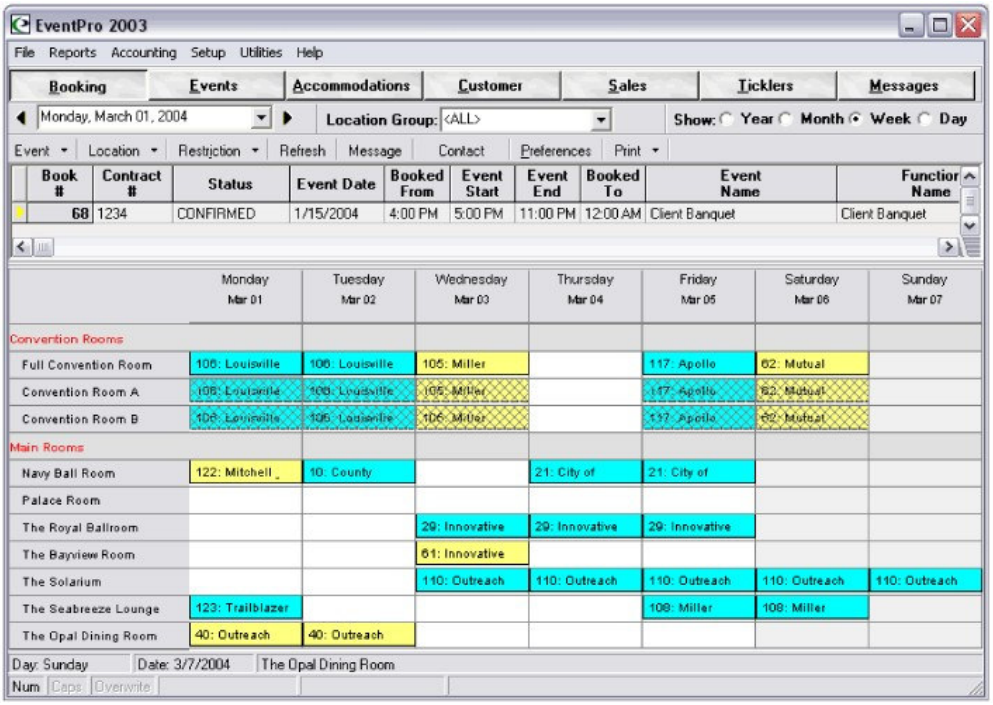
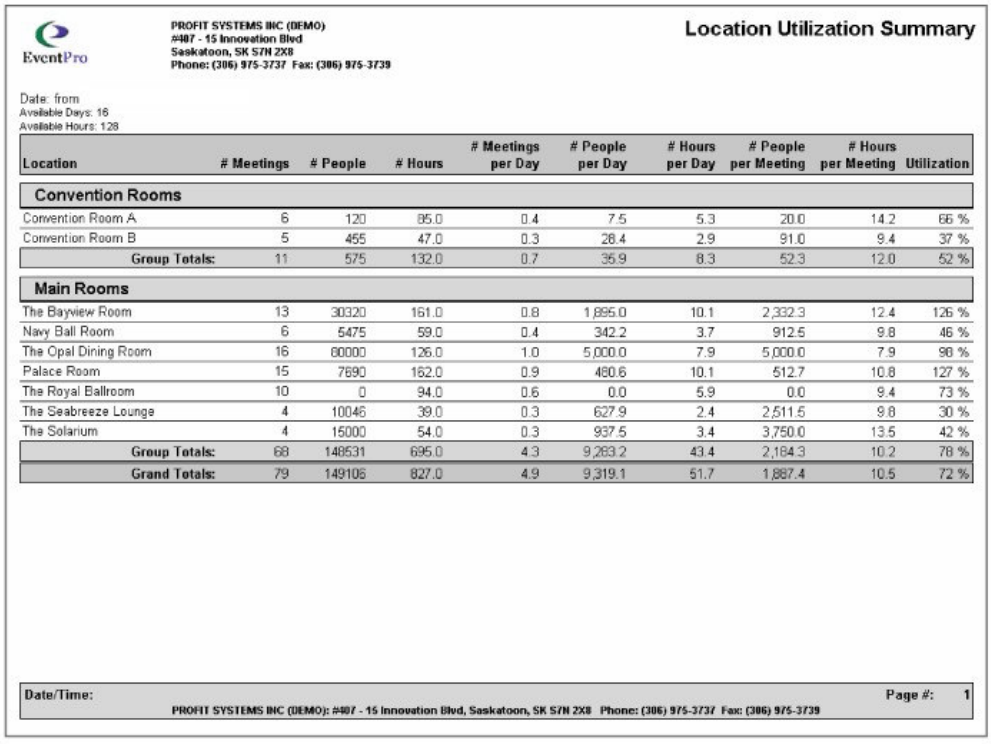


Figure 2.9 Example of the use of colour by EPP

In contrast to how offensively these systems present the data they hold, EPP provides an excellent reporting tool. It offers the user both a wide selection of reports, as well as allowing them to customise their own. Figure 2.10 show a sample report from EPP. The reports are neat, easy to read and offer the user portable, well set out and useful information.



EventPro

PROFIT SYSTEMS INC (DEMO)
#407 - 15 Innovation Blvd
Saskatoon, SK S7N 2X8
Phone: (306) 975-3737 Fax: (306) 975-3739

Location Utilization Summary

Date: from
Available Days: 16
Available Hours: 128

Location	# Meetings	# People	# Hours	# Meetings per Day	# People per Day	# Hours per Day	# People per Meeting	# Hours per Meeting	Utilization
Convention Rooms									
Convention Room A	6	120	85.0	0.4	7.5	5.3	20.0	14.2	66 %
Convention Room B	5	455	47.0	0.3	28.4	2.9	91.0	9.4	37 %
Group Totals:	11	575	132.0	0.7	35.9	8.3	52.3	12.0	52 %
Main Rooms									
The Bayview Room	13	30320	161.0	0.8	1,895.0	10.1	2,332.3	12.4	126 %
Navy Ball Room	6	5475	59.0	0.4	342.2	3.7	912.5	9.8	46 %
The Opal Dining Room	16	80000	126.0	1.0	5,000.0	7.9	5,000.0	7.9	98 %
Palace Room	15	7690	162.0	0.9	480.6	10.1	512.7	10.8	127 %
The Royal Ballroom	10	0	94.0	0.6	0.0	5.9	0.0	9.4	73 %
The Seabreeze Lounge	4	10046	39.0	0.3	627.9	2.4	2,511.5	9.8	30 %
The Solarium	4	15000	54.0	0.3	937.5	3.4	3,750.0	13.5	42 %
Group Totals:	68	148531	695.0	4.3	9,283.2	43.4	2,184.3	10.2	78 %
Grand Totals:	79	149106	827.0	4.9	9,319.1	51.7	1,887.4	10.5	72 %

Date/Time: Page #: 1

PROFIT SYSTEMS INC (DEMO): #407 - 15 Innovation Blvd, Saskatoon, SK S7N 2X8 Phone: (306) 975-3737 Fax: (306) 975-3739

Figure 2.10 EPP Sample Report

Event Master also has a change in feel once you get away from the mass of data. It too can supply the user with various reports, and also has other useful outputs, for example an event manager may require such as badges. See Figure 2.11 Event Master Output and Printing. It is logically set out, and draws on users' previous experience of computers to navigate around each section. (The concept of clickable expanding options is comparable to what is used in most directory viewers). With a unique built in badge designer this section offers what the user may want in a logical, straightforward fashion.

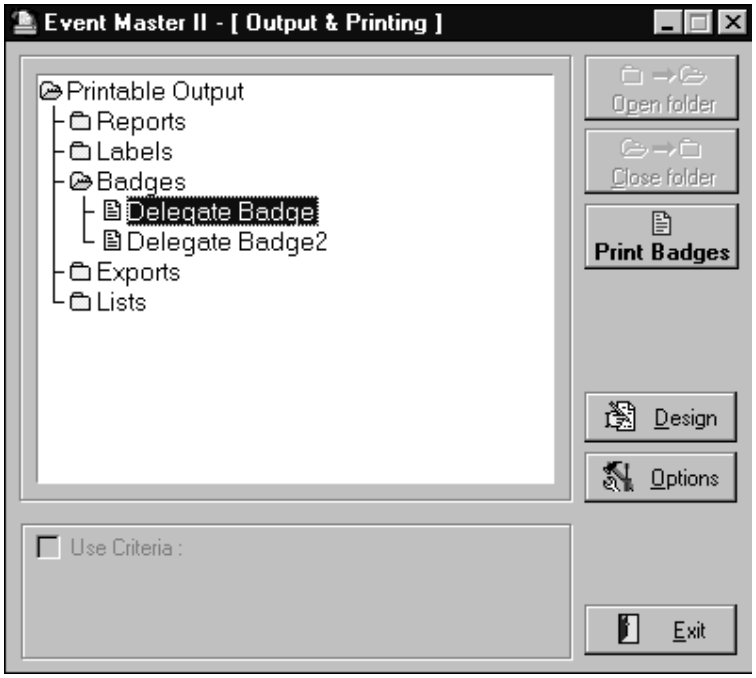


Figure 2.11 Event Master Output and Printing

To conclude, event management software currently available struggles to cope with the data needed to organise events, and tries to provide too many features that may only cause confusion to their users. However, they have several nifty features that can greatly assist the event manager – especially the reporting facilities.

2.11 Summary

The future of learning in the classroom is in a more varied set of interactive and lecturing techniques. Computers are beginning to be recognised as the future for more effective and stimulating learning.

Electronic performance support systems are gradually infiltrating learning in organisations, as the need is primarily for the user to complete a given task – not to learn how to do it. It is widely acknowledged that a company that successfully implements an effective knowledge management and performance support system that empowers the employee to be more task focused, will be unstoppable. Jobs will be fully supported by integrated systems providing workers with all the information and support they require, freeing up their minds to provide a higher quality of work.

Event management is not something that can be taught, but an effective performance based system can greatly assist students, both by providing them with information and supplying a medium through which they can organise events effectively.

The task of actively encouraging students to organise their events in a safe and intelligent way and then facilitating this activity is one that has not yet been implemented

3 Current Student Event Management Advice

3.1 Introduction

Every University has a collection of clubs and societies that run their own events. Advice given to them generally takes the format of paper based information booklets, providing general information on any policies that may affect students running events, and also gives any recommendations for how to run them.

This section seeks to identify the positive and negative aspects of the advice that is currently available to Students, which may influence the design of this system.

Some Universities provide their information for students online, and some of these are investigated below.

3.2 Southampton Students' Union [11]

Southampton provides a very brief outline of advice, detailing only what is essential for students to know in terms of their policies and procedures. There is nothing specific, and it only details the bare essentials of the next steps that need to be taken – for example, to contact the events manager.
There is one positive design aspect about it:

- The concept form forces organisers to think about all aspects of the event – not just the budget, before planning begins. An event that has good figures on paper does not necessarily mean it will be good.

3.3 Nottingham Students' Union [12]

The student safety advisor has taken on this task in order to provide support to organisers, and particularly emphasis the safety aspects of an event. It takes the form of a workbook and supplement which uses a "fill in the blanks" method so students can also record their actions. This has many good features:

- It immediately outlines the responsibilities of the students – the common law duty and in turn the responsibilities of contractors and other involved parties.
- It not only lists the contact details of relevant personnel, but also what they are useful for.
- Budgets insist on an inclusion of a 10% contingency and a 10% damage payment should they be necessary.
- Event checklists are included for both the planning stages and on the day.
- There is a section allocated for drawing a site plan or seating plan – I would like to use the capabilities of the system to store building layouts for viewing.
- The workbook gives space for students to add their own notes but only in a few places.
- The supplement booklet gives details on the additional hazards and how to handle them and also recommends the extra planning days required to deal with them appropriately.

3.4 Bath Students' Union

The societies team is currently in the middle of putting their advice into online documents. (Advice on event management is not yet complete). Each society used to be given a

[11] Event Support Guide [Online] Available at http://union.susu.org/events/eventsupportguide.pdf (accessed November 12 2003).
[12] University of Nottingham Students' Union :: Get Involved! [Online] Available at http://www.su.nottingham.ac.uk/index.php?module=ContentExpress&func=display&ceid=612 (accessed November 12 2003).

booklet containing all the information they needed to run effectively, of which one of the sections was on events. There are many positive points about the booklet and also some which need to be considered for the online tool:

- Information is informal – in so called "student speak". This may help relate to users more easily but could also be construed as annoying.
- The guide is very thorough and lists a lot of information. I think it would be read only once, and then not used again. To combat this, the system could have a novice and expert environment to ensure that the system caters for users of all abilities.
- It is a thorough information provider. This is going to be the main attraction of the system – users need only go to one place to get everything that they require – or links to everything they need.
- It includes an expert guide to union policies and catering requirements. This could be expanded to include a guide to all the union forms, as they too are not always easy to understand.
- It has an informative hints and tips section which could be expanded to include a fact finder and FAQ section, complete with real stories to add to the 'reality' of the system.
- The advice is very bland. For example, it may say, "find a venue". An online system can help users to benefit from the experience already gained by other societies. It can give examples of venues used by others before, and the associated costs as well as any problems that were encountered.

In addition to the paper based system, one member of the society's team begun to put together relevant information for students regarding events. The project was not completed, but it would simply print a report of instructions (that may or may not be relevant to students) depending only on the type of event. A sample report from the system is included in Appendix C. As the system was not completed, it has never been used by students so it is not possible to evaluate it formally.

3.5 Summary

Currently, there is no application available to students to help them organise their events. There is a lot of paper based material which may influence parts of the design of this system, but information is mainly presented to students all at once, as a bulk of ideas and advice.

4 Requirements

4.1 Introduction

This section is critical to the progress of this project and seeks to establish a firm set of requirements on which to build a design. The requirements are derived from many sources, including the literature review, the users themselves, and influenced by several key application principles such as usability and accessibility.

Whilst many requirements can be obtained directly from the literature review, several of the points raised were not conclusive and were therefore discussed with the potential users. It is crucial that these requirements are as comprehensive and clear as possible, to provide the design with the strongest possible foundations.

It is unavoidable that some of the requirements will conflict with others mentioned in this section, so these are detailed here, making way for the design to resolve them.

Some of the requirements are relevant to more than one section. For example, a requirement may be discussed in both the literature review and usability requirements, but are listed in both places for completeness.

4.2 Requirements resulting from the Literature Review

These requirements summarise the findings concluded in the literature review about the application's content and design. The numbers LR 1 to LR 16 represent the numbers of each requirement which will be referred to in the rest of this project from here onwards. They are divided up into several sections:

4.2.1 System Requirements

The system:

LR 1) Must be accessible 24 hours a day, 365 days a year to allow users to access its service any time they wish.

4.2.2 Design Requirements

The system:

LR 2) Must present a friendly environment and use sociable language.
LR 3) Must be designed around the tasks associated with it, not the data it portrays. This requires further investigation as seen in section 4.4.
LR 4) Must provide users with just enough information at just the right time to allow them to complete their tasks.
LR 5) Must be as self explanatory as possible, and require a minimum amount of training in order to be used.
LR 6) Must provide expert and novice modes to allow different types of users to all interact with it in the most advantageous way possible.
LR 7) Must provide students with a comprehensive set of resources to enable them to complete every task they will face.
LR 8) Must provide a comprehensive set of advice and guidance to support students through the whole process of organising and running an event.

4.2.3 Functional Requirements

The system:

LR 9) Must provide students with training, exercises and tips to help prepare them for the event they will run, teaching them how to handle difficult situations that may arise.
LR 10) Must facilitate the storage of graphical files such as maps and venue layouts.
LR 11) Must allow users to produce custom designed reports.
LR 12) Must facilitate knowledge sharing by providing a means for students to discuss any issues or questions they have for each other.
LR 13) Can help put students at ease by providing a way for them to share stories of things that have gone well or badly, as a way of encouragement and preparation.
LR 14) Must provide a comprehensive help system both to support students using the application and also to answer any questions they may have about organising events.
LR 15) Must encourage students to complete an evaluation of their event that can be put back into the system to support students in the future.

4.2.4 Resources Requirements

The system:

LR 16) Must ensure the accuracy and reliability of the data as much as possible.

4.3 Additional Administrator Requirements

The Students' Union societies' administrator had some specific requirements of her own that she wanted to see the system adhere to. Outside of these, she was merely interested in a system which could support students in organising events. These are addressed below:

4.3.1 Functional Requirements

The system:

Ad 1) Must contain an event diary that students can enter all their events into. The order which they were entered must also be displayed.
Ad 2) Must encourage students to provide feedback on their event and store this information for future reference.
Ad 3) Must only require a very low level of administration.

4.3.2 Content Requirements

The system:

Ad 4) Must encourage students to organise their events in a responsible manner with specific attention paid to:
  i. British and European Laws
  ii. Relevant required licensing regulations
  iii. University and Students' Union policies.
Ad 5) Must allow the administrator to change the resources and advice given to students.

4.4 Issues Uncovered in the Literature Review Requiring Further Research

Several of the topics discussed in the literature review could only conclude that the answer to them in regard to this application was dependant on the opinions of the potential end users. This section describes the issues to some extent, which section 4.5 will then refer back to and explain how they will be addressed.

Issue 1) Scheduling

Section 2.5.1 introduced the concept of using formal scheduling tools when completing projects and also events. However, as this section describes, due to the demands of student life, formal techniques may not always be appropriate for students, although some form of scheduling is vital.

Issue 2) System Features

Several of the systems reviewed in section 2.10 seemed to be overloaded with too many features with the potential to cause the user a lot of confusion. The question arose as to whether students would prefer this all-inclusive nature for an application, or would rather keep the guidance system as simple as possible and use packages they are already familiar with to supplement their performance.

Issue 3) Methods

Text books are very good at looking at the underlying theory of project and event management (see section 2.4), but as a starting point for a student system, what we really need to know is how students do things. From this, it can be decided if what they are doing is good enough, or if they need to be encouraged to follow different techniques.

Issue 4) Evaluation

The administrator has requested it, sources referenced in the literature review repeatedly deem it essential, but it is important to know how students view this. If they do not hold event evaluations and hand-over reports in high regard, incentives and encouragement may have to be applied for them to participate in it.

Issue 5) Demand

No matter how much the administrator believes in it, or how much evidence there is that this system would be highly beneficial to students, if they themselves do not see the benefits, it would probably not be used. If there is no demand for such a system by the student population, or there is no motivation for them to use it, it will fail.

4.5 User Research

To truly understand the requirements of the system, it is necessary to understand its potential users. That involves identifying the tasks carried out by students to achieve their goals, and how they complete them, as well as ascertaining what their current difficulties and expectations are. A well known method for this is task analysis, which is used to understand what users are trying to accomplish, along with why and how they are achieving their goals.

Task analysis of event management in this volunteer sector is very difficult to monitor, due to tasks being completed between lectures and other commitments, on an ad-hoc basis when the student has time available. Another complexity is that each event is also fairly unique, and it is impossible to find two events exactly the same in every way to compare them.

Given these constraints, there are techniques that will help understand the users' approach and hence help develop a set of user requirements.
The plausible techniques available are:

- Interviews - users are questioned on a one-to-one basis.
- Focus groups - questions or situations are put to a group of individuals for discussion.

- Questionnaires – a quick and easy technique used to gauge the opinions of a large number of people.

Focus groups are an excellent way to get a broad set of views from users, and they encourage them to bounce ideas around and discuss differences of opinion. Preferably, focus groups should be run for each group of users individually. However, it very difficult to distinguish such groups as most people have organised an event of sorts, but of a whole variety of types and scales. Additionally, to get the optimum results, a minimum of 3 focus groups per user group should be run in order to the greatest number of opinions.

Unfortunately, due to the time and cost constraints of running so many groups, as well as the difficulty of finding users from all the categories willing to partake, the focus groups have been limited to just one – consisting of students who have organised a range of different types of events within the Students' Union and University context to be able to utilise the time in the best possible way.

However, in order to augment the results of the focus group, a questionnaire will be used to ask some of the same questions to a wider audience. The questionnaire is discussed in section 4.5.2.

4.5.1 The Focus Group

Although a complete task analysis is impractical, the focus groups will be used to establish how things are done, and how people differ in their approach to the task of organising an event. One way in which this can be done is to give participants a set of sub-tasks of an example event to get a grasp of how they approach the event and how they group tasks together. Another is to chose a small task, such as "get hold of helium for balloons", and give them resources, such as a computer, telephone directory, and telephone, and see how they go about achieving their goal.

Both of these approaches will be used to get a better grasp of how students organise themselves. The rest of the focus group seeks to resolve the issues described in section 4.4 and seek to identify a set of user requirements.

The focus group content has been based on Krueger and Casey's (2000) recommended format, consisting of 6 sections, which also follows their "do's and don'ts" of focus group design.

The focus group chair's full set of notes and the materials to be used can be found in Appendix D.

Opening

The chair introduces them self and lays out the instructions for the focus group, encouraging the participants to express their opinions as much as possible and reassuring them that their answers will be kept confidential.

Introduction Questions

These questions introduce the user to the topic and are generally easy to answer.

1) Why did you get involved in organising events?
2) How many people do you usually work with on events?
3) Who are they? Friends? Other Society members? And in what capacity? A fellow organiser or an advisor?

By their selection into the focus group, it is already known that every participant has organised at least one event so question 1 probes into their reasons for it. This is

important to help truly understand the user and is a key component of task analysis – which is one aim we are striving to achieve.
One of the requirements resulting from the literature review is that knowledge can be shared between users. Question 2 goes a step further to consider the realm of event sharing – if there is a need for more than one user to be able to access each event. Question 3 verifies this information: For example, a student may consider that they work with a sabbatical officer when organising an event, but that may be in an advisory capacity – and only the organiser would require access to that event in the system.

Transition Questions

Transition questions are more in depth, and start to look at some of the main issues. They help the participants gain momentum before reaching the key issues.

4) Consider that you are about to start working on a new event. What are the stages you go through, starting with the event idea through to the event itself, and so on. Write your ideas on paper which will be discussed as a group shortly.
5) Have you ever come across any exceptions to this list of stages?

This section was used to start the students thinking about how they break down the task of "organising an event" into the smaller, yet still high level sequence of tasks they go through. Question 4 gives them the opening and closing tasks to ensure they understand the question and they can then write their answers individually on paper, in preparation for a group discussion. This follows Issue 3 from section 4.4. Section 2.4.2 of the literature review showed that there are exceptions to the planning stages found in so many key texts, so question 5 is asked to see if these students (with such a variety of event experiences) have ever come any exceptions. This will hopefully be a good indicator as to whether these so called "exceptions", are indeed exceptions or in fact the norm.

Key Questions

These are the questions that drive the study, that find answers to the issues required to progress into the design phase of the system. They are divided into sections to help the focus group progress smoothly and are discussed below.

6) When completing a project as a group, do you set in stone when and who will complete each task? Do you use formal methods such as a Gantt chart to record this?

7) Exercise: How do you organise yourself: The set of cards in front of you are a set of tasks associated with running a sports quiz with a raffle. Imagine you have thought of these. How would you group them together / record them in a filing system? What titles would you assign to each section?

These two questions pick on Issue 1 of section 4.4. Question 6 aims to answer it, in order to establish scheduling requirements for the system, whereas question 7 is more practical. It uses a realistic scenario (used by a project team to run this event last year) to see how individuals tackle the situation. The grouping of tasks will show how many groups the students like to divide things into, as well as realistic headings for each one. (This information will be useful when grouping information and teaching points for the system itself.)

8) In front of you is a computer connected to the internet and your university file space, a telephone, local business directories, and some paper. The task you are given reflects a typical task you might complete if you were organising an event. You can use anything on this table to complete this task and record your findings. [You can do anything, call anyone, look at any website etc].

Although it is impossible to carry out a complete task analysis as previously discussed, this exercise aims to simulate parts of the process to understand how users organise their events. One at a time, each participant will be sat at the table, and asked to complete a typical task they would have to do if organising an event properly. The computer would also have common programs stored on it, such as a web browser, and Microsoft Office. As discussed in Issue 2 it will be interesting to see what the students automatically use to record their information.
The aim of this is to see what resources they use, how they record their findings, and how many different sources they look at before stopping.

The tasks that will be used are:

- Book a transport for 50 people to see a show in Bristol
- Organise a country put tour route
- Organise some decorations for a dinner in the Claverton Rooms
- Book a restaurant for an arts show meal for 35 people
- Book a balloon modeller to appear at a department ball

9) What skills do you think you need to be a successful event manager?

10) What support do you currently receive? Would you like to have more support?

Section **2.5** of the literature review discussed some of the skills related to project management, and question 9 seeks to expand and verify this list. The aim of this is to decide what key skill areas the system should provide training for.
Before progressing into talking about a system to support event managers, question 10 gauges the opinion of the users to see if they think more such support would be worthwhile.

11) Consider that a system was designed to support student event managers like you, -rate these functionalities / features in order of importance.

12) If summaries and organiser evaluation/hints/tips of previous events were available, would you use them?

13) If asked to, would you complete a short evaluation of your event for the benefit of others?

The answers from question 11 will be used to complete the system requirements, and also to construct a set of implementation phases - the features considered the most important being implemented first. Following issue 4, questions 12 and 13 delve into the user's perception of the usefulness of evaluation, and if the users would complete one themselves. Discussion around question 14 should hopefully determine how easy the system should make writing an evaluation. This will also make an impact on the requirements.

Ending Questions

Ending questions draw the focus group to a natural conclusion, and reflect on the topics previously discussed.

14) All things considered, would you be seriously interested in using such a computer based system? Or simply as a resource facility?

15) Do you think that a system would be beneficial to students who had never organised an event before?

16) Do you have anything else you would like to add about organising events, or a potential events support system?

Question 14 and 15 aim to judge the students' final opinion on whether such a system would be useful to them, and hence conclude whether there is demand for it or not. Question 16 is an invitation for the participants to bring up anything else they feel has not already been covered.

Closure

This final section is a way of concluding the focus group, and asks participants if they are interested in further studies regarding this project.

17) Would you be interested in commenting on the design of this system?

18) When the system is finished, would you be interested in testing it?

4.5.2 Questionnaire

Due to the constraints mentioned previously, the focus group will be augmented by a questionnaire. It will ask similar questions to those discussed in the focus group and will hopefully result in a broader understanding of both experienced and non-experienced potential users.

Targeting the participants

Experienced users, that is, students who have run events within the constraints of the Students' Union that will be targeted are:

- Rag members
- Society event officers
- Students who took the event management unit as part of the Sports Science degree.

Non-experienced users who have an interest in events are very difficult to find, so fellow final years from the computer science department will be targeted, as well as personal friends.

Designing the questionnaire

Questionnaire design is an art in itself, so Oppenheim's book (1992) was used to construct a questionnaire that was easy and simple to answer, that extracted the best possible information from the partakers.
As mentioned before, they were designed around the questions in the focus group, and an example of both the questionnaires for the novice and experienced users can be found in Appendix E.

4.6 User Research Results

In combing the results from the focus group and the questionnaires, a good analysis of what users are looking for in a system is possible. The interesting results are discussed below and are split into sections by topic area. The complete statistics resulting from the questionnaires can also be found in Appendix F and the transcript of part of the focus group found in Appendix D.

4.6.1 Overview of participants

As predicted in section 1.5.2.1, students who consider them selves to have no experience of organising events are not easily found. Of a total of 42 people who filled in the questionnaire, 33 of the claimed to have event management experience.
The five focus group volunteers were chosen because between them they had a wide variety of events behind them. two had worked heavily with the University in organising student events, another two had great knowledge of charity run events (through Bath Area Rag) and also through organising Freshers Week – the biggest Student run event on the calendar, and the final participant was active in finance work, working with

societies to budget their events effectively, as well as having involvement with arts events.

4.6.2 Support

The experienced organisers were asked in the questionnaire what support that had received when organising an event, and only two felt that had not received any. Friends and family supported 21 of the students, whilst 20 and 19 received support from Student Union and University staff respectively.
From the specific events mentioned in the questionnaire, it appears that much of this support comes from the lecturer of the event management unit (as part of the sports department) but comments from the focus group suggested that such help was supplementary to their events or concerned specific areas only.
In the case of Freshers Week, a central University committee assists the event managers in parts of their roles, whilst for the fireworks displays, the University Safety Office, and Facilities department are crucial to its organisation.

4.6.3 Organising themselves as individuals and as groups

The vast majority of both novice and experienced students stated in the questionnaire that a form of paper is their preferred way of organising themselves. The nature of a paper system means that it is transportable and is something tangible that is not so easily lost. These feelings were reflected in the focus group as well as the feelings that you can store everything you need in a paper file and always have it in your hands to refer to. Some of the focus group participants commented that they would not trust a computer as much – because it is less in your control and is perceived to have more of a mind of its own.

When the students were asked how they organise themselves as part of a group, the majority of novice students would assign people tasks as the project progressed, whereas a third of those with experience preferred a more advanced method of scheduling tasks. The focus group also showed that this very much depends on the context and the situation. Each event will be different and organised differently. The fireworks display, although headed up by a student team has certain tasks delegated to University staff. Another condition is the people who are organising the event, and if the person in charge trusts their co-organisers to complete a task competently.

The exercise in the focus group to observe how they grouped tasks together showed varying results. Four of the five participants grouped the tasks by subject, such as “Publicity” and “Venue”, favouring simplicity. The remaining participant structured their tasks on a “to do list” basis, such as “on the day”, “logistics”, “promotional material” and “content”. What was made clear by everyone following this task was that they would prefer the flexibility to organise things how they wanted – just like they can currently control a paper based system.

4.6.4 Event System Features

The following table shows the combined results of both questionnaires, showing how each potential feature was ranked.

Rank	Feature
1	Resources
2	Guidelines
3	Summaries of previously run events
4	Checklist
5	Budget help
6	Event evaluations
7	Ideas lists

8	Quote Management
9	Sample events (for training)
10	Help on filling in official forms
11	Fun training

Table 4.1 Questionnaire results ranking the importance of system features

The focus group allowed for discussion around each of these features and their potential advantages and disadvantages as summarised below:

Resources

The resources were seen as advantageous as they decrease the amount of searching needed to find potential suppliers/venues etc, but it was suggested that the list should be kept short to give students ideas, whilst encouraging them to look for themselves. The group agreed that this list should also include names of people to talk to for advice or further information.

Guidelines and Checklists

The discussion began with one person saying that accurate guidelines were crucial for first time event organisers, or people who are not natural event organisers. There was a feeling however that advice given by the computer could be ignored, but this was seen as the potential for any case. For the participants, despite the feeling that they would be told what to do by a computer, they liked the fact they these guidelines could be ignored, but were a very useful reference to check themselves against.

Handover reports and Event summaries

These were the most discussed features of all, with everyone agreeing that these were extremely useful and almost crucial to the event manager's tasks. The potential that any application could encourage this, or made this task easier was considered excellent.

Quote management

Following the point made about users preferring to organise themselves in paper based systems, managing quotes can require many associated documents, which can all be stored in a file together. However, for the purposes of event sharing this was seen as a useful feature as long as all the changes made to the quotes were highlighted with when the change was made and who made it. Once again, participants commented that this also depends on the type of event, and the type of quotes that need to be gathered.

Help

All kinds of help or training on teaching organisers how to do things well were taken badly. The participants did not feel that they needed any such help, and didn't think that first time organisers would appreciate it either.
However, frequently asked questions and simple "How to…." resources, such as "How to book a band" and "How to fill out the £200+ event form" were seen as useful but should also point people in the direction of Student Union staff if they required more help.

Lists of ideas were seen as useful, but would require an administrator to monitor them so that no unsuitable ideas were given out.

Final Judgement

Finally, both the questionnaire and focus group participants were asked if they would consider using such a computer based system and the result was particularly positive. Even though some people felt that they would not like some of the features, they would use the system for resources and checking they had completed all the required tasks regardless.

4.7 User Requirements

Following the results from the user research, a set of resulting requirements can be listed.

The system must:

- User 1) Give users flexibility in how they organise their events
- User 2) Allow users to feel in control of their event planning
- User 3) Assure the users that regular backups are made so in the occurrence of a problem, their work will not be lost
- User 4) Require minimal data entry
- User 5) Display when any changes were made to the event and by whom, for events being organised by more than one person
- User 6) Encourage and facilitate users in completing an event evaluation and handover report
- User 7) Provide a "How to…." help facility. For example, "How to fill out the £200+ Event Form"
- User 8) Provide students with contact details of where they can get further help if it required
- User 9) Provide resources, guidelines and checklists of things to do for event organisers to use and compare themselves against
- User 10) Provide a quote management facility
- User 11) Provide a list of suitable event ideas

4.8 Use Cases

Event management by its nature will include many people – suppliers, contractors, bar staff, volunteers etc but the two positions significant to organising events within the students union, are the students, and the societies administrator.

4.8.1 Student use cases

The student has one principal, all inclusive use case – to organise an event. As discussed in section **2.2** in the literature review, this is a complex task composed from many inter-linking tasks that must be completed for a student to achieve their objective. Following discussion with the administrator and students in the focus groups, the use cases can be broken down into several smaller ones, which in turn can be broken down even further. Not all of these use cases are specifically linked to organising an event but are included here to show what background knowledge most students enter the event management situation with.

1) Understand how the Students' Union operates

Be elected as a society officer
Complete compulsory society officer training

- Meet the administrator
- Complete basic training on how to use the Student Unions' website (bathstudent.com).
- Learn basic procedures regarding budgeting, finances, booking rooms on campus, etc.

2) Understand the proposed event

If this is a repeated event from the previous year or semester:
Discuss the feasibility of the event with the administrator and the rest of the society committee
Get feedback from previous event organisers

OR

Gather ideas
Discuss the feasibility of the event with the administrator and the rest of the society committee
Get feedback from event organisers who have organised something similar

3) Get the go-ahead to do the event

Complete the "£200+ Event Form" (Found in Appendix A)
Get help on filling it in from the administrator
Get the form signed by the relevant sabbatical officer and society committee member

4) Organise the event

Plan every aspect of the event
Get contracts signed
Complete a risk assessment
Clarify an event plan

This step is where most of the work takes place. "Plan every aspect of the event" will change depending on the event being run and will include sections such as: Book a venue, book travel, book entertainment, set-up a communication protocol, write evacuation plans etc. It would be impossible to detail every section, as there are an infinite number so some common examples are broken down below. Several of these also overlap – for example, a venue will often insist that you use their own catering.

Book a venue
- Search for suitable venues that are available
- Visit them to assess suitability
- Negotiate a price and conditions of use
- Gain booking confirmation in writing
- Pay deposit
- Meet with the venue

Book an entertainment (e.g. magician, DJ, karaoke)
- Gather quotes
- Make a decision
- Confirm performance times and set lengths
- Get their contract signed by a sabbatical officer
- Pay deposit (if required)
- Obtain a copy of their Public Liability Insurance
- Pay fees in full

Security
- Call firms recommended by the Students' Union
- Compare prices and decide which will be used
- Meet and discuss requirements
- Carry out a site visit with the chosen company if necessary
- Agree and confirm times for security coverage
- Pay deposit
- Get their contract signed by a sabbatical officer
- Pay fees in full

Risk Assessments
- Read up on how to complete an event risk assessment
- Read risk assessments and handovers of previous events
- Write a draft risk assessment
- Discuss it with the Student Activities Safety Advisor

- Update risk assessment and ensure that it is followed.

5) Run the event

Put the plan into action
Setup the venue
Brief volunteers / staff / security

6) Completion

Ensure the finances are completed
Complete an evaluation report on the event
Send thank you letters to appropriate parties.

4.8.2 Administrator Use Cases

Following discussion with the administrator, establishing use cases here for a potential system is very difficult. The points outlined below discuss the administrators two main responsibilities in the events area:

1) Deliver society officer training

Related to the first student use case, the administrator has deliver training to those involved with societies.

- Deliver bathstudent.com training
- Introduce the students to basic procedures regarding budgeting, finances, booking rooms on campus, etc

2) Assist event organisers as required

Similar to the fourth student use case, just as students have to organise each aspect of an event which will vary dramatically between events, so the administrator has to assist them with the tasks that they complete - when required.

These may include:

- Discussing the feasibility of new events with students
- Assisting in student to staff negotiations (e.g. in discussing the hire charge for the student nightclub)
- Helping students understand and fill in all the required forms
- Solving any problems that may arise.

4.9 Usability and Accessibility Requirements

Although the topic of usability is not the key focus of this project, as the resulting system will be used by humans to support them in completing the task of organising an event, usability principles must be adhered to. They are summarised in Appendix H and must be considered in both the design of the system, and whilst testing it.
Similarly, there are clear guidelines set out by the W3C web consortium regarding making web pages accessible for all users. These are summarised in Appendix I, and will also be considered during the design and testing stages of this project.

4.10 Non-functional Requirements

Sommerville (2001) outlines three main areas of non-functional requirements that must be considered whilst designing an application. In such an informal setting, many of these have not yet been discussed explicitly but are more crucial than any functional requirements because non-compliance with them may render the entire application unusable. There is no specific guideline that recommends that “a system must respond

within 5 seconds to any request made of it" for example, so the basis of these non-functional requirements comes from the requirements already specified, as something to aim for. Some of these requirements that come under this section have already been discussed, such as the usability requirements, so they will not be repeated.

4.10.1 Product Requirements

These are the requirements which specify how the application should behave.

The system must:

Non-func 1) Not make wait users longer than two seconds for a page to load on a 56K connection unless they are informed of the delay and the reasons for it
Non-func 2) Be accessible to users of all browsers of all screen resolutions. Note: The use of the system on small screen display such as a PDA is not required
Non-func 3) Take every step to avoid failure which must be monitored and minimised.

4.10.2 Organisational Requirements

These are derived from the customs and policies specified by the customer's domain.

The system must:

Non-func 4) Only be available to students of the University of Bath
Non-func 5) Provide information which conforms to both the University's and Students' Union policies.

4.10.3 External requirements

These external requirements are derived from external influences to the application and its development.

The system must:

Non-func 6) Abide by the Students' Union privacy policy
Non-func 7) Meet the requirements of the Data Protection Act 1998
Non-func 8) Work alongside the Students' Union's current web site and link to the information it holds where relevant
Non-func 9) Operate in an ethical manner.

4.11 Conflicting Requirements

In some cases, it is simply impossible to meet the precise demands of the requirements outlined above. At some point a decision will have to be made over which requirement is the most important or a compromise must be reached.

4.11.1 Data overload verses storing links to external data

Storing information as straight data presents it to the user immediately, on screen. If information is linked to, there is a lesser need for maintenance as it is hoped it would be kept up to date by the author. However, it cannot be assumed that external data will be updated, so it will have to be monitored and checked. The links themselves will also have to be checked to see if they are still active.

4.11.2 Better features verses the ease of maintaining the system

Time in the Students' Union is short, and previous systems that have been so called "self-maintainable" have proved otherwise. The Students' Union new website promised to be an all singing all dancing system which required very little human intervention yet 12 months on, a full time website administrator has been employed for it.
Some features of the system that may be very useful – such as a contacts list and venue listings – contacts that previous students have used, along with their comments and evaluation of them. These contacts would have to be vetted to ensure fairness and accuracy, and whilst the system could do a lot of the work automatically it is hard to imagine that no user intervention would be required.

4.11.3 Database speed / efficiency verses a Higher level of normalisation

Whilst normalising a database is essential, it is often difficult to decide to what extent it should be done to ensure that it is understandable to the administrator (should it need to be changed) and what effect it has on the speed of the system. Pulling data from one table will often be quicker than pulling it from three different ones joined together to get the same information. Normalisation is also important to avoid redundant and/or out of date information.

4.11.4 The need for excessive data entry – what is relevant and what is not

Whilst many of the systems analysed in the literature review included comprehensive data entry handling, it is very near impossible and often pointless to achieve a paperless event management office. For example, copies of suppliers' insurance certificates may be posted or faxed to the event manager for their reference, which are likely to be stored in paper files. A line must be drawn somewhere in the student context, as how much information is essential to the event management team, and what can be stored elsewhere, or different programs used. For example, students who were part of the focus group said that they used a spreadsheet to track ticket sales for large events. Hence where is it necessary to draw the line. The other point is that students are unlikely to have access to the system during the event itself, so a portable device such as a PDA or paper, will be required to list things like the event plan.

4.11.5 User Control verses Structure

To ensure that the students complete their events properly, and to minimise the risk of them missing out a crucial part of the planning, some form of structure must be instilled, and the way they plan their events dictated in some way. On the other side of the coin, the students want to feel in control of what they are doing, and not feel bullied by the system into doing what it expects.

How these conflicting requirements are resolved, is discussed in the design section.

4.12 Coding Standards

As this prototype is only being developed by one person, and it is only a prototype with the code unlikely to be re-used, coding standards could initially be construed as not holding significant importance.

However it is good practice to implement coding standards for consistency, understanding, and there is still a chance that the code may be re-used.

4.12.1 Database coding standards

Table names will be written in capital letters such as "GUIDELINES" except where the table is one that in practice links to others, whereby the "to" will be written in small letters, such as "GUIDELINEStoCRITERIA".

The following prefixes will be used to represent the type of field the table is storing:

Prefix	Field type
i	Integer
si	Small integer
v	Varchar (text up to 255 characters which does not store white space)
c	Character
d	Date
t	Text (for text bigger than what can be stored with a varchar)
pk	Primary Key

Table 4.2 Database field types

In addition, the 'pk' prefix can be merged with the 'i' prefix to denote an integer primary key – thus giving the prefix 'ipk'.

4.12.2 Code Storage and File names

The file space which will store the code for the application will be split into several sections:

- Pages: will hold the scripts generating the actual pages of the application itself
- Functions: will hold the checking scripts and include files referenced within the pages
- Buttons: will hold the image files of the buttons used in the application
- Images: will hold all the other images required by the interface.

4.13 Limitations of the Requirements Gathering

Whilst far from perfect, this requirements section seeks to identify as many of the requirements as possible, with particular importance on the thoughts of the users. After all, they are the people this system is designed for, and who will be using it.

The main difficulty associated with this section was completing sufficient user research. All the students are volunteers within their society, and not even a request from the society's administrator offering free food and drink to them all persuaded any to partake in a focus group.
Because of this, only one focus group was held which is far from ideal so the questionnaire was used to supplement some of the results. Running more focus groups would have been likely to portray a wider set of results and give a greater understand of user thinking.

As previously mentioned, a complete task analysis was simply not practical due to the nature and schedules of the potential users so activities as part of the focus group attempted to fill this gap.

4.14 Summary

Despite the limitations outlined, this requirements document provides an adequate set underlying requirements on which the design can be based. It has fully explored the main areas for requirements – those derived from the literature, the users, and other key principles.

Those requirements causing conflict with others have been identified which the design section will seek to solve. These requirements also provide a method by which the system can be tested following its design and implementation.

The design section which immediately follows, will use these requirements to create a suitable design for the event management support application.

5 Design

5.1 Introduction

Based on the requirements document, this design lays out the structure of the application, its front end and the underlying database.

Fundamental design decisions are discussed firstly, along with the issues presented in the requirements that are resolved before the design is considered.
Once again, the potential users are involved here, to give their thoughts on the structure and front end of the application and the resulting adjustments and additions explained.

The requirements section lays out the needs of the complete event management solution, and whereas it will eventually be necessary to implement all or most of the features discussed, it is simply not possible to do this within the time constraints of this project. The design therefore focuses mainly on the subject, which has not been seen in any existing applications - that of providing students with the relevant information and guidelines they need to organise their events. Facilities such as managing a contact list, and a “to do” list are seen in many pieces of software already – Microsoft Outlook for example.

5.2 Specific design decisions

Following the requirements specification, there are decisions to be made regarding the foundations of the systems design. That is, how the requirements are to be met in the best way possible.

5.2.1 The underlying system structure

Following the findings of the literature review, and specifically Section **2.10** looking at the design of existing systems, the structure will be modular based. This design was the most logical way to not only display data, but also the least confusing for the user to understand. However, the titles for these sections need to be chosen carefully, and the links between them made obvious to avoid the confusion that was seen with some of the existing solutions.

This not only applies to the overall system, but also to each event. Event procedures and resources will also be presented in a logical unambiguous way to make the system as easy to use as possible.

5.2.2 Task centred design

Following on from the previous point, the system must be designed primarily with the task in mind – and not only around the data that is required for it. Section 2.9.2 discusses the dangers of this, and presents the concept of the performance zone. The design will strongly consider this, and aim to provide the right information at the right time to the user to make the task of organising an event as straightforward as possible. See Section 5.4 below.

5.2.3 Scope of the system

The amount of in-built features a system of this nature should have is discussed in section 2.10. It argues that having more features does not always mean a system is more useful. The scope of this system will be limited to reflect this, and will advise users to utilise packages they are familiar with to complete tasks – such as a spreadsheet to help write a budget. The results from the focus group (completed as part of the requirements

section) show that students are willing to use such packages already – and do it automatically when presented with a task.
This system will not try to mimic the facilities that such packages can provide.

5.3 Resolution of conflicting requirements

It was seen in the Requirements Section that several of the requirements conflicted with each other. This section seeks to determine how these conflicts will be solved, or which requirement has to be ignored.

5.3.1 Data overload or links

Summary of the conflict: Whether to 'hard code' information into the system (hence creating a greater maintenance requirement) or store links in the system to reference external information. For example, whether to store the details of how to write a risk assessment document, or store a link to a comprehensive document on how to do this on an external site.
Resolution: All documents internal to the Students' Union and the University will be referenced by the system as a link. It is likely that these links will be kept up to date and can be trusted; however the administrator must have the facility to check these links periodically, and be able to alter them, edit them, or remove them and replace them if necessary. The administrator should also be alerted if any content on these pages changes.
This facility should also apply to external documents. If there is a good resource available with suitable guidelines for students to help plan their events, it would be a waste of time to recreate this information for the purpose of inputting it into the internal system.
All linked documents should be scanned periodically for accuracy. In line with Requirement LR 16 the user should be alerted to any significant inaccuracies found.

5.3.2 Better features verses ease of maintaining the system.

Summary of the conflict: Whether a feature such as an intelligent contacts section (listing contact information, and evaluation of the contact by the person that used them) is worth the administrator's time in validating it.
Resolution: The system should be as self developing as possible so if a user enters a set of contact details, they should automatically be entered into the central contacts database if the user wishes. However, as seen from the user research in section 4.5, this is not one of the top priorities for students, so it is unlikely that this feature will be implemented given the time required.

5.3.3 Database speed / efficiency versus a higher level of normalisation

Summary of the conflict: Whether the database should be normalised fully if the speed of pulling data from it was made slower.
Resolution: The database behind the system will potentially hold a very large amount of data and if it is not normalised correctly, will make it very difficult to maintain. As seen already, it is key that maintenance is made as quick and simple as possible. However, it is important that data can be pulled from the database as quickly as possibly and techniques such as field indexing should also be used to help this.

5.3.4 The need for excessive data entry – what is relevant and what is not.

Summary of the conflict: How much information is necessary to be stored in the system, and how much can be left on paper.
Resolution: Only the key information needs to be input into the system, that is, details concerning the quotes and whether or not relevant tasks have been completed or documents gathered. For example, if the contract has been signed - all that is necessary to store is a simple "yes" if they have been done.

5.3.5 User Control versus Structure

Summary of the conflict: To what degree the student should be pushed through the system to make sure they organise their events properly, or if they should be given total control.
Resolution: The system shall deliver advice to students mainly in two sections – those guidelines that must be considered before a supplied is confirmed, and those they can be dealt with after the confirmation. This will hopefully not overload students with too many things to do, but will still give them a feeling of control.

5.4 System flow

The use cases specified in section **4.8** along with the findings of the focus group (despite its limitations) can be grouped and used to develop a list of tasks to instil into the systems design.

The key stages identified (in section 4.8.1) were:

1) Understand how the Students' Union operates
2) Understand the proposed event
3) Get the go-ahead to do the event
4) Organise the event
5) Run the event
6) Completion

Step 1 is not related to the task of organising an event so can be ignored here. According to the results of the requirements focus group, step 2 may or may not be completed here, so the system must not insist on it. Evaluations of previous events and lists of ideas should be made available and the user alerted to them when considering a new event.
Step 3 is nearly always completed as part of step 4, so the system must allow for this. The information required for Step 4 can be broken down into distinct sections according to the type of planning element, which can in turn be segmented further depending on what information is required at what time. This helps to put into practice a performance zone as discussed in Section 2.9.2, and one example is that of the information required to book a supplier. Several things need to be done before the supplier is confirmed, and several others need to be done afterwards.

A number of documents will be needed to help run the event in step 5, for which the system can advise and help produce, and then step 6 finishes off each event. The system can assist the user into producing an evaluation then the information can be fed back into the system to help event managers of the future.

In analysing these steps, they can be converted into a system depicted as follows:

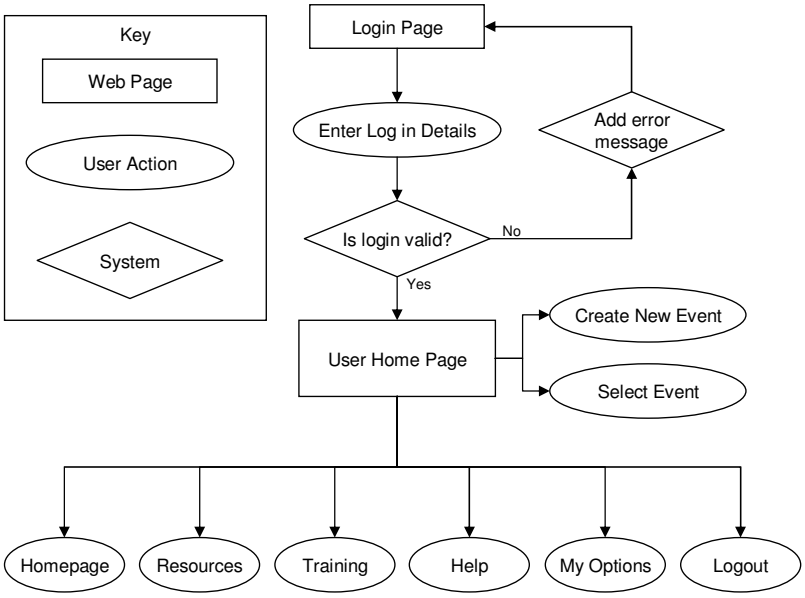


Figure 5.1 System flow chart of the proposed system

Note: This diagram also includes provision for training and help as these are stipulated in the requirements. The detailed designs of the rest of the system can be found in Appendix G.

5.5 User Interface Design

The design of the user interface is crucial to the project, because if it doesn't present the user with information to complete their tasks in a logical and useful way the entire system will be useless. To ensure that the system is useable its design will follow the W3C web consortium guidelines and the usability requirements as featured in Appendix F.
A series of mock-ups will be used initially to test the design with several potential users to check that it meets their expectations.

The key design areas will now be discussed, along with the rationale behind each of them. Established design patterns will be followed where applicable to help create the most usable application possible; these are summarised in Appendix J and are referenced by number in the following breakdown.

5.5.1 System Design Structure

Mock-up number: all
Incorporates design pattern number: 1

This agrees with the decision made in section 1.2.1 above, to structure the system in a modular way. A navigation menu is set on the left hand side which can be clicked on to enter each respective section.

5.5.2 Login page

Mock-up number: 1
Incorporates design pattern numbers: 2,3,4

The key to the login page is to introduce the system, allow existing users to login, and direct new users (who must be University of Bath Students) to signup to the site. If you

are not a verified user, the site has nothing further to offer you (except to view the system tour) so no navigation is required here.

The login area requests a username and password to enter the site, which the user is asked to enter before clicking on the action button “Login”. If they fail to enter both a username and password, they will be immediately presented with a message box instructing them to do so. If the username and password combination is invalid, the user will be told what the problem is at the top of the box, and instructed to try again. Reasons why the error has occurred will be listed below the box. All error message text will be displayed in red to alert the user, and symbolise a problem. In this instance it is not appropriate to initiate the registration automatically if an incorrect username is entered. Users *must* be Bath University Students as previously mentioned, and the easiest way to check this is for the registration page to have access limited to this group only.

In addition, the login area offers a link for the user to get help on signing in, and a link to an opportunity to be emailed their password if that have forgotten it.

Users will often use a shared machine to access the system, as very few students have a personal machine on campus. Therefore, provision has not been made to store passwords locally but web browsers often have this feature built in to them, so users accessing the system from their own machines will be able to store their access information using this method if they wish to.

When the user has logged in successfully, they are taken to their personal homepage and a welcome message “Welcome [user’s first name]” is displayed.

5.5.3 For all other pages

Now the student has been validated and logged in to the system, the full system capabilities can be offered to them and personalised.

Several of the features are present across all pages of the system as follows:

Menu bar

The menu bar is clear and easy to understand, and sits on the left hand site of the page. It provides links to the homepage, and the main sections, and to the “logout” action.

Navigation bar

Incorporates design pattern number: 6

A navigation bar informs the user of where they are currently located in the system hierarchy and allows them to retreat to any stage of it by selected on the name of that level. It is displayed in the colour burgundy to contrast with the systems other colours and stand out whilst being aesthetically pleasing.

Title bar

Incorporates design pattern number: 5

This title bar is found across the top of every page in the system (including the login page) for consistency and visual effect. Wherever the bar is displayed, the user can be sure that they are within the application and by clicking on the logo they can return to the homepage whenever they wish.

Message bar

The message bar provides a personalised welcome message when the user reaches the homepage, and can provide other useful information when the user is in other parts of the system. This may be a reminder as to what section the user is currently viewing, or a reminder of the name of the event they are currently planning.

5.5.4 Personal Homepage

Mock-up number: 2
Incorporates design pattern number: 2

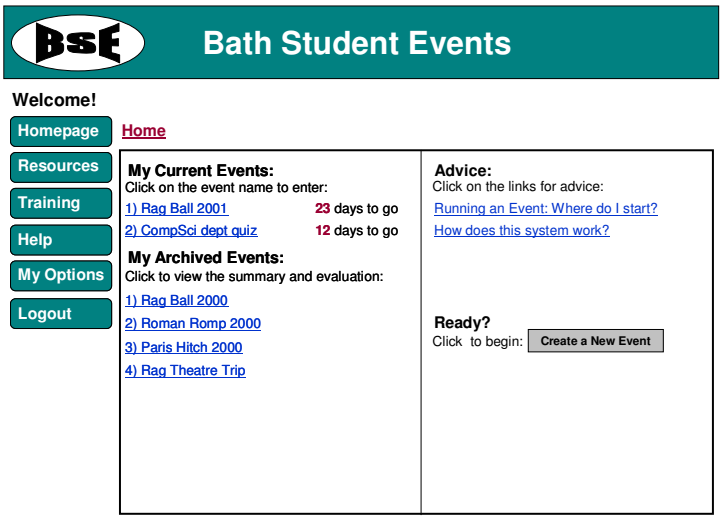


Figure 5.2: Personal Homepage Layout

The personal homepage aims to welcome the user to the system and provide them with relevant information and links. The key feature of the system for every user is assistance with planning events so this page lists each of the users' events as well as the number of days to go until that event.
Additionally, should the user wish to view the details of the events they have previously organised, they can access them from here, as well as being able to set up a new event.

Relevant advice is also linked to, for example, "How does this system work" if the user has not accessed the system before; along with any news relating to the system or announcements made by the site's administrator.

The links on this page, that is, to the events, advice and announcement links are displayed in blue underlined text, as would be expected by the user for a link taking them to a different part of the site – without consequence. However, they will not change colour when they have been visited, as they will be visited multiple times and hence will always appear a different colour. It may be suggested that the other links should change colour when they have been visited, but I have decided against this to maintain consistency.

5.5.5 Resources index

Mock-up number: 3
Incorporates design pattern number: 7

The resources index lists all the major sections of resources available as well as the sub sections within them. A description of each top level section has also been included to provide the user with enough information about it, to decide if it is what they want to look at. If the user clicks on a top level link, it will take them to a similar page to this one, listing links and descriptions of the level 2 sections.

Furthermore, if the user is looking a page of a level 2 section, the navigation bar will display exactly where they are. This means that the user can back track to one level higher. Each level 2 page will also display links to other level 2 relevant sub sections to provide the user with additional navigation should they want it.

5.5.6 Event planning wizard

Mock-up numbers: 4, 5
Incorporates design pattern number: 8, 9

The planning wizard requires several pieces of information to setup an event plan for the user. This is split into two stages – the event's 'personal details' – its name and date (if known) and details concerning its type and contents which the planning wizard needs.
The wizard allows the user to back track to stage 1 if they wish (via the navigation bar), but there is no means for the user to see what stages there are in the entire process as there are only two, which was not considered enough to warrant such a feature.

When the user confirms their event details at the end of the second stage, they are sent to a page which asks them to confirm the criteria they have selected, to try and catch any last errors the user may have spotted. If they choose that these are correct, they will see a processing screen while the system sets up their event. This processing screen will inform them of the task that is currently taking place, and will then redirect them to the event plan screen when it has finished.
However, every effort shall be made for the system to be quick enough for such a processing screen not to be required.

5.5.7 Event plan guide

Mock-up number: 6

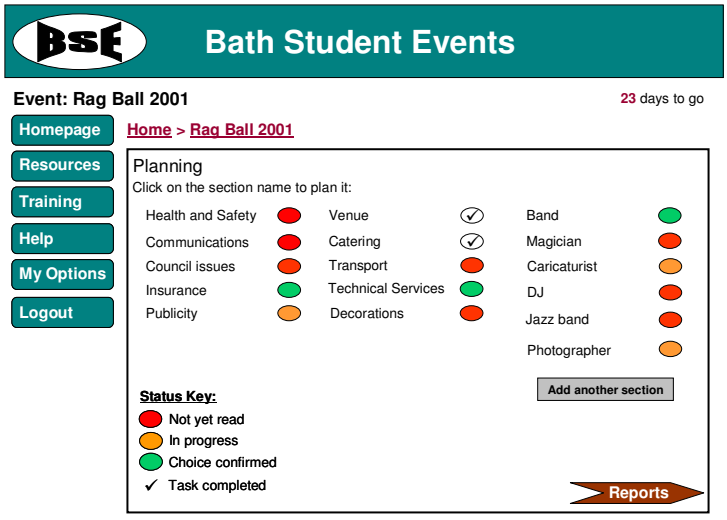


Figure 5.3: Event Plan Guide Layout

This page allows the user to manage their event planning in small portions. The sections and their content relevant to the event (as determined by the decisions made by the user during the wizard) are listed on this page with an associated circle indicating the status of each section. This is for the user to easily see what sections they have been completed and what stage those that haven't been completed are at.

A key is also included at the bottom of the page for clarity, although the colours used are similar to those found in everyday life – of red, orange and green. (That we see in traffic lights for example).

5.5.8 Event plan section

Mock-up numbers: 7,8
Incorporates design pattern number: 10, 11

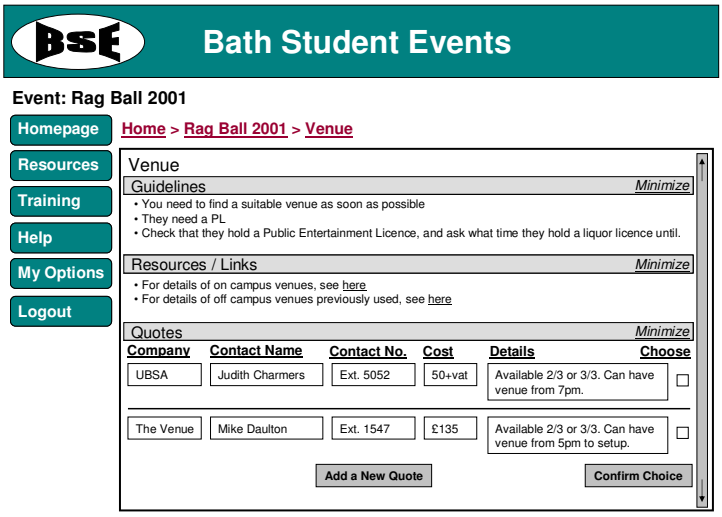


Figure 5.4: Event Plan Section

This is the page which provides the users with the information they require. Contrasting colours were used for the rows of the table that depicted the headings in order to make them stand out and for the user to focus in on which particular part of the page they were looking for. This is an adaptation of the design pattern 10, as using contrasting colours on every row would have detracted from its readability in this instance.

In order to make best use of the space available, the quote section is presented in a compact yet logical way, with ample space for students to input the key details required. (This relates to 5.3.4 – discussing the amount of data entry required. Note that this is only a sample set of fields. For now it is not relevant to investigate what these fields should be).

After the user confirms what quote they have decided to keep, mock-up 8 shows what the screen would look like. The previous sections have now been minimised, but the user can expand them if they want to, to look back at the information. The confirmed quote is now shown, along with the final "to do" list related to this item. This relates to 5.3.5 – which specifies that information should be kept appropriate to the user, and appropriate to the stage which they are at.

5.6 User Testing

To check that the key design features and the mock-ups are what the real users expect, user testing will be carried out to obtain some feedback on them. This questioning will also encourage feedback and ideas from the participants.

The participants will be presented with the screen mock-ups, and asked for their opinions on its layout and features, as well as being asked what they would expect of it following

various actions. A PowerPoint presentation of the mock-ups in Appendix K will be used to carry out the tests.

5.6.1 The users

As this is only a brief survey to gain user feedback on the design work so far, I believe that four users will be enough to carry out the testing on. Two of the participants will be chosen at random from the focus group participants, and two will be potential users who were not involved for that stage, to try and get a broad and unbiased set of opinions.

5.6.2 The questions

The following questions were asked to the participants for each PowerPoint slide they were shown.

1. What would you expect to happen when each item/button/link was clicked on?
2. Is there anything that strikes you as annoying about this page?
3. Is there anything that is not obvious as to its meaning or action?
4. Would you like to see any part of the page displayed differently?

5.6.3 The results

All of the links, buttons and items presented were seen as obvious to each person questioned, although it is important to note here that each user has at least 3 years experience of using a computer.

The only exception was that none of the participants thought that the logo would take you back to your homepage – they believed that it was simply a graphic. Other comments regarding expectations of the system mainly concern navigation through it, and are discussed along with the rest of the comments by section below.

Personal Homepage

The “days to go” function was instantly noticed as a useful feature being both informative and as a good “reality check”. However, it was also very strongly felt that this could become exceedingly irritating. Organising events can be stressful, so individuals must be given the ability to turn this function off at any time. Displaying the date of each event alongside it would also be nice to see.

Quick reference to previous “archived” events was also picked up on as being useful, but one which could get very long. The initial suggestion was to simply have a link to a separate page containing this information. However the group concluded that listing the 4 most current archived events, with a link to a page with all their archived events listed would be the most useful compromise.

Navigation wise, the wording of the motion to go to start creating a new event was seen as patronising and childlike.

Resources Index

This was only commented on as having a very web-like appearance but this was however easy to understand and every participant understood where they would be taken if they clicked on any link.

The Wizard

Despite omitting a “proceed to stage 2” button on the first stage, this was seen as user-friendly, and an ideal way to collect the required information. The only suggested design alteration was to the question asking how many people would be involved in the event. One participant explained that users may feel trapped by such a question, and that asking them to enter the number of people they thought would attend would be friendlier.

The Planning Index

This was liked for being "task complete" – showing the user a complete picture of what they were trying to do but the way that the status of sections were displayed was considered flawed. Not only would this mean very little to someone who was colour blind, but could also be represented better with symbols for example.

Content wise, users liked that they could add their own sections, but would also want to delete some which they did not see relevant.

Sample Planning Section (Venue)

This was perceived as the core of what the system could do, and there were lots of ideas bounced around of extra features and functionality that this could hold. The most popular of which was to make it all more encompassing, and allow the "to do" lists to be seen at this stage too, as well as allowing the user to add their own "to do" actions.

Although initial discussion whether "quotes" should be included or not, was not favourable, the group concluded that a link to storing quotes on a separate page would be better because it was possible there would be a lot of them. It was also noticed that the system didn't allow users to delete or update the details of the quotes they had entered, which would be essential to any working system.

The only confusion here was how to return to the main planning index page, and although the "bread crumbs" were understood, it was not seen as enough. A button would be a well placed addition to this page.

5.6.4 Resulting changes

The user feedback prompted the addition of several further requirements. Many of these requests concerned extra features and functionalities which are out of the defined scope of this prototype so will not be considered any further at this point.
Below are the extra requirements that are relevant within the designed prototype:

The system must:

- Design 1) Allow the user to turn off the "days to go" feature at their discretion
- Design 2) Contain links to archived events that the student has previously run
- Design 3) Not depend on colour alone to symbolise the status of a planning section (that is, if it has not been started yet, is partially finished, or completed). This is also necessary for accessibility requirement number 2.
- Design 4) Allow the user to delete any sections of the event plan as they choose
- Design 5) Allow the user to add their own planning sections
- Design 6) Allow the user to add their own "to do" actions to the checklist
- Design 7) Allow the user to delete and update their quote information
- Design 8) Contain extra navigational features for clarity

If this application was developed further, it would be necessary to consider all of the additional functionalities requested and investigate them further.

5.7 Database design

The design of a database to cover every element of the tasks wanted for this project, including help, introductions to the system, all the resources and guidelines etc would be a huge task. This design only covers the areas outlined for the prototype.

The database design focuses on the system's most important function – that of being able to create an event plan for students to use, giving them the information and resources they need to complete the associated tasks. The event plan will vary according to the type of event, and features of it. For the purposes of the project, the resources and

guidelines themselves are not part of the scope, so the information contained in the current Bath System (as described in section 3.4) will be adapted for this application.
Having said this, the sections that have been written will be designed to allow further development work should this be required at a later date.

The database follows the coding standards set out in Section 4.12.1. It is designed such that it is easy to understand and maintain as well as being correctly normalised.

It is important to remember that there are countless ways to write a database. The method I chose focuses on the task the user has to complete, and the data that requires, then works backwards from that central task to get hold of the required source information.

It is easiest to understand the database by dividing it into 3 sections:

- Administration
- Master data storage
- Event data storage

5.7.1 Administration

This section contains 3 tables that hold the user and event information.

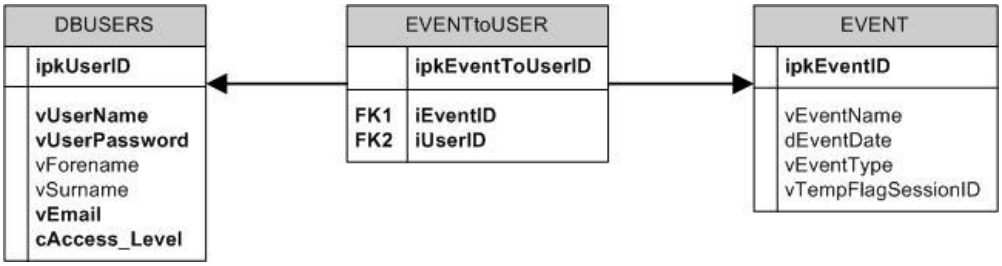


Figure 5.5: Administration Tables

DBUSER stores the key information on each user.
The vEndYear field will store the students' graduation year, so that once they have graduated they can be removed from the system. The user information is the only critical information required by the system, and the user will be able to alter it at any time they wish. This ensures compliance with the Data Protection Act.
The two principle uses of this table are to validate user logons, and provide the administrator with their details if they ask a question via the system.

The EVENT table stores key information on each event.
The dEventDate field is not only used for alerting users of how many days they have left to plan the event, but could also be used to link to the Students' Union's event diary. The idea of storing what type of event it is, is to eventually provide context sensitive help to the users.

The EVENTtoUSER table provides a link between users and the events they organise. This table allows the system to support event sharing.

5.7.2 Master data storage

The tables here are used to store the data that provides the guidelines, resources, and checklists relevant to the user depending on the criteria of the event. The data is split into sections, such as "Venue", "Entertainment" and "Safety".

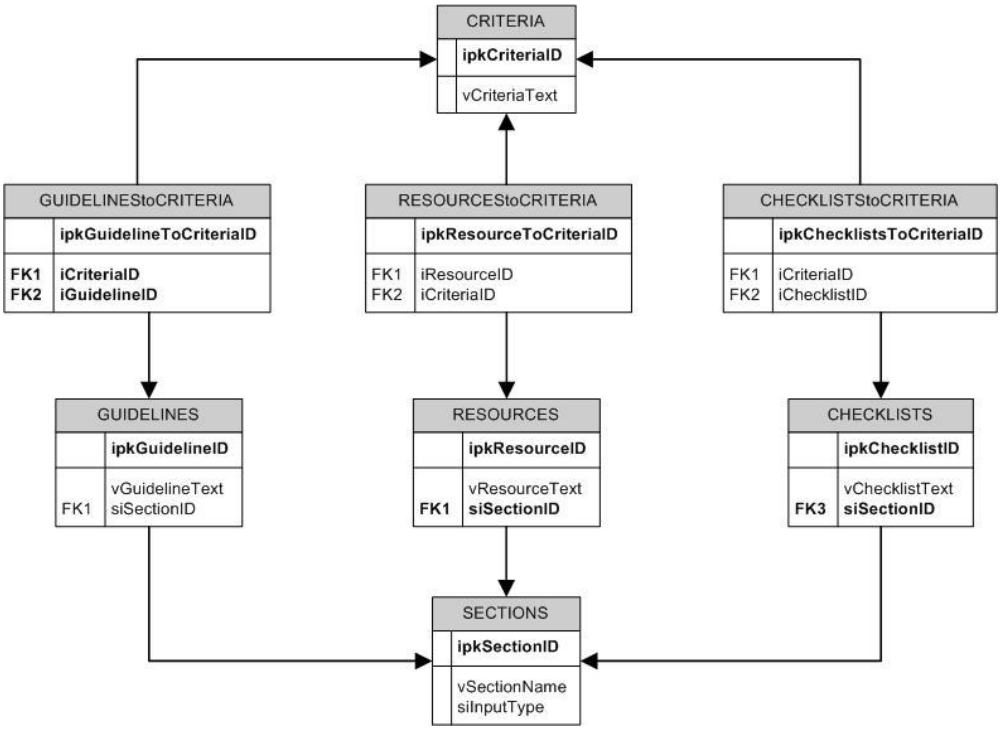


Figure 5.6: Master data storage tables

Each guideline belongs to section and is only relevant to an event if one of its criteria is listed in the event set-up.
For example: The guideline “Posters for your event can be placed on Students’ Union notice boards around campus if they adhere to the poster policy” is only relevant to events meeting one or more of the following conditions:

- The event is being organised by a Students’ Union society
- The event is being held on campus.

These are known as the criteria, of which there are currently 17.
This guideline should only appear in the Publicity Section.

CRITERIA stores the identification number and description of each of the criteria.
Guideline Id numbers and their text along with the section id number that they relate to, is stored in GUIDELINES. Then the GUIDELINEStoCRITERIA table stores the links between the guidelines and the criteria.

In relation to the previous example, if the guideline given had the Id number 44, and the criteria Id numbers were 5 and 6, the record stored in the GUIDELINEStoCRITERIA table would be:

iGuidelineId	iCriteriaId
…	…
44	5
44	6
…	…

Exactly the same structure is in place for the Resources, and Checklists.

5.7.3 Event data storage

Given that the master storage data holds the information itself as well as the criteria and sections that each belong to, the system can then populate these event data tables for each event.

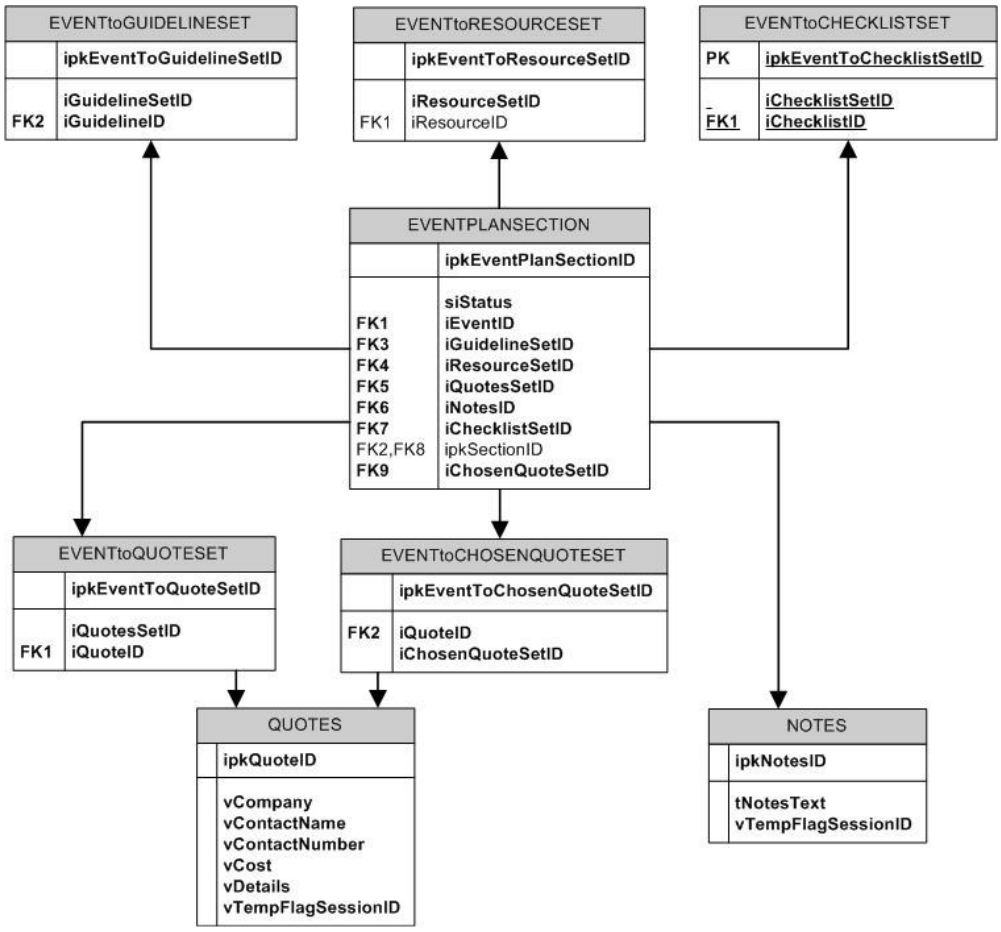


Figure 5.7: Event data storage tables

Each event has a number of sections which will always have a set of associated guidelines, and may also have any number of the following: Resources, Checklists, Notes and Quotes.

A central table, EVENTPLANSECTION stores the information about every section of every event. It stores the identification number of the event it belongs to, and then the identification numbers of all the information associated to that section.
The guidelines related to the section are grouped as a set, with one GuidelineSetID - stored here. The table EVENTtoGUIDELINESET then stores the links between the GuidelineSetID and the GuidelineID.
Similarly, the other sets of information are stored with link tables to the master data in the same fashion – that of the Checklists and Resources.
The quotes are also stored in the same way, although the QUOTES table stores user created data.
The notes for each section are stored slightly differently. As each section can have at most one “note” attached to it, there is no need for the ‘set technique’ to be used. Instead the NotesID field refers directly to the NOTES table.

If any quotes are gathered for a particular section, the user can confirm whether which quotes they are using. As they may be more than one of these, a table EVENTtoCHOSENQUOTESET stores the links of the chosen quote set to the QUOTES table – with the quotesetid being stored in EVENTPLANSECTION.

5.8 Summary

This design section has outlined all the work necessary to write the prototype for this system – including page layouts, the database design and the rationale behind them.

The inclusion of user input hopefully helped to eliminate any design flaws, and will help to ensure that the proposed system meets the users' expectations.
Once again, the amount of user feedback was not ideal, and holding more focus groups with a wide variety of participants would have been likely to yield even more beneficial results.

The following implementation section will seek to apply this design work suitably to produce a suitable prototype.

6 Implementation

6.1 Introduction

The complex part of implementing the prototype system was in its design and structure, making way for a relatively uncomplicated implementation.
Several features were chosen for complete implementation while other functionalities were implemented partially to augment it and help the user visualise the further potential of the system. This section discusses the decisions surrounding the implementation and gives examples of how the resulting application looks.

6.2 Techniques Used to Ease the Implementation Process

To limit the work required to implement the prototype and to prevent wasting time, several techniques were used to make the work required as painless and quick as possible.

Firstly, some of the functions such as the code used to calculate the number of days between 2 given dates, was adapted from free code available online. These functions are clearly referenced in the code.

Using style sheets not only cuts the number of formatting tags required in the main body of the code, but also helps web pages look consistent. The W3C web consortium also strongly recommends their use.

Finally, the use of include files. Functions and parts of the page design, such as the title bar and the message bar, were stored in include files so they could be used multiple times on different pages. This also promotes consistency, as any changes to these parts of the system need to be completed only once.

6.3 Key Implementation Decisions

The following decisions were made during the coding stage of this project to ensure the most effective implementation with the time and prototype constraints.

Sessions

Sessions are the PHP method for handling logins to a system. PHP typically stores its session information in a cookie, but to ensure that the use of the system was not affected if the user did not have cookies enabled; the session identification number was also passed in the pages' web address.
For example:
http://www.bath.ac.uk/~ma0car/project/pages/wizard.php?PHPSESSID=1b6f2fgsd4462

Screen Area

To present an aesthetically pleasing system to all users with a whole variety of screen resolutions, the decision was taken to implement the system using strict areas. With a screen resolution of 600 x 800 the application would fill the screen, but at higher resolutions it would simply appear slightly smaller.

This format was chosen to avoid the hassle of coding a flexible screen structure, but the optimisation of screen usage is desirable to many users and should be investigated if this application was to be developed further. This is discussed in the Testing section.

6.4 Database implementation

The database was implemented exactly as specified within the design section. Two mySQL front-end applications were used to simplify the process of setting up the database, and later helped to test it.

6.5 Application Front End Implementation

A simple text editor was used to help write the PHP code which built the front-end of the system. It had the facility to recognise PHP commands and format the code to aid readability.

6.5.1 Page design template

The pages of the prototype were each based on one central template shown in Figure 6.1. The only slight deviation to this is the login page, which does not have the Navigation Bar, where the page content stretches instead across the full width of the page.

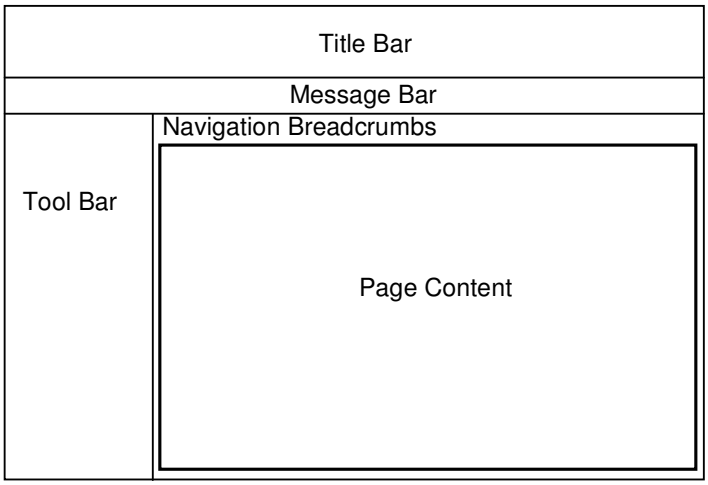


Figure 6.1 Design Template

6.5.2 Description of the Code

The complete code listings can be found in Appendix O listed in alphabetical order but this section will describe a brief overview of how it fits together.

Looking back to Figure 6.1, the partitions directly relate to several of the functions:

- Header.php contains the code for the title bar
- Messagebar.php contains the code for the message bar
- Toolbar.php contains the code for the toolbar, taking the text to be displayed from the page it is part of.

The navigational breadcrumbs will be different for each page so they are set up within the page they form part of.

Checklogin.php is used throughout the application to check that the user is logged in with a valid session and to ensure that the session variables are set correctly so they can be used effectively by the system.

Login.php creates the login front page that directs the user to homepage.php (the homepage) when they have successfully logged in. The homepage welcomes the user,

and incorporates the functions contained in date_eval.php to tell the user how many days they have until their event will take place.

To set up a new event, the two stages the user goes through are newevent.php (Stage 1) and wizard.php (Stage 2) which then submit their information to setupplan.php that creates the data required for the event.

The part of the system that allows the user to organise their events are planevent.php that lists the section of the event which in turn links to plansection.php displaying the information related to the chosen section. This page has the most functionality associated to it in the prototype and is supported by the following functions to complete its objectives:

- UpdateNotes.php: updates, deletes or creates a new note as requested.
- UpdateQuotes.php: updates the information the user has entered in the quotes section, deletes one if required, and confirms a quote if the user requests this.
- PostchoiceSetup.php: sets up the page with the relevant "to do" list included.
- UndoChoice.php: cancels the user's choice of quotes and returns them to the previous page.

6.5.3 Sample Run Through

The following figures show some sample screenshots of a typical session using the application.

Figure 6.2: After the user has successfully logged in, they arrive at their personal homepage listing the events they are currently associated with, along with the option to set up a new event.

Figure 6.3: To set up a new event, the user must complete a two step wizard. This screenshot shows the first of those steps.

Figure 6.4: Once the user has set up their event, they are presented with a planning index listing all the things they must consider for their event. Here, "First Aid Arrangements" has been selected and the user is displayed the guidelines and resources for this section. The system also invites them to add any personal notes to this point, and any quotes they wish to store.

Figure 6.5: For the venue section, after the user has entered all the quotes they wish to, as well as any personal notes, they can choose one of their quotes. This screen then shows them the final organisation points which they must complete to finalise this plan.

Figure 6.6: As part of the help section, users are invited to email "Charlie" [13] with any question they may have. The system recalls what the last event was that the student looked at, and inserts the required information although the user can change this if they want. Clicking on the "Send Email" button will pass the question on.

[13] "Charlie" is the name of the society's administrator. This approach helps the system stay informal. However, the application currently submits the questions in an email formal to myself.

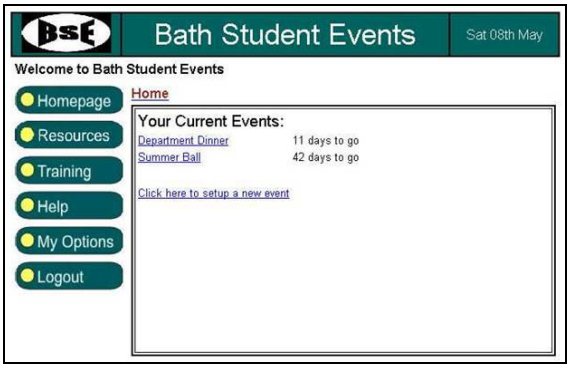


Figure 6.2 Personal Homepage

Figure 6.3 Event Setup Wizard

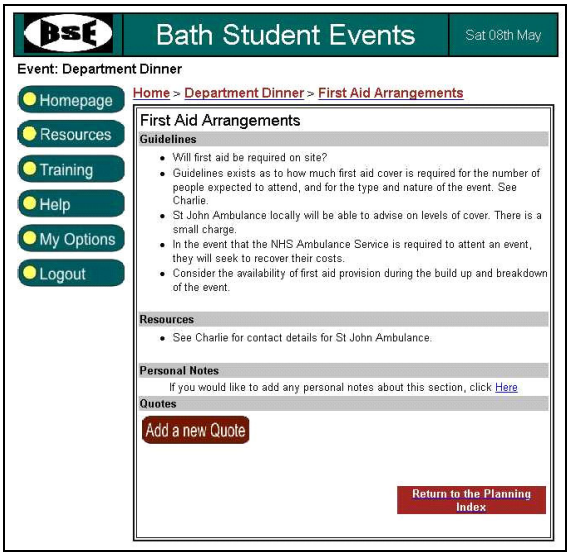


Figure 6.4 First Aid Arrangements

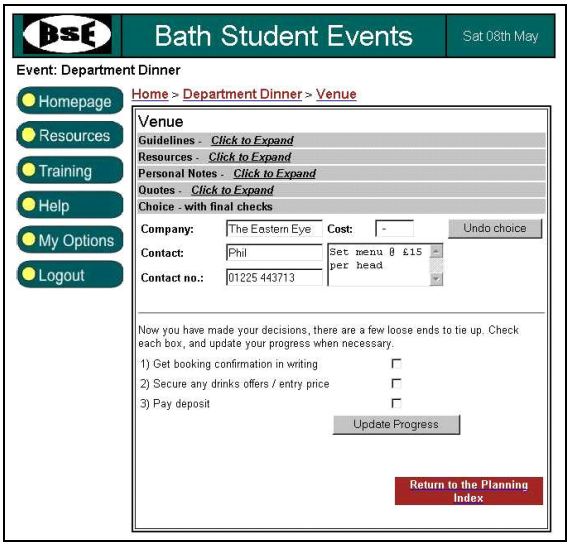


Figure 6.5 Venue – after a quote has been confirmed

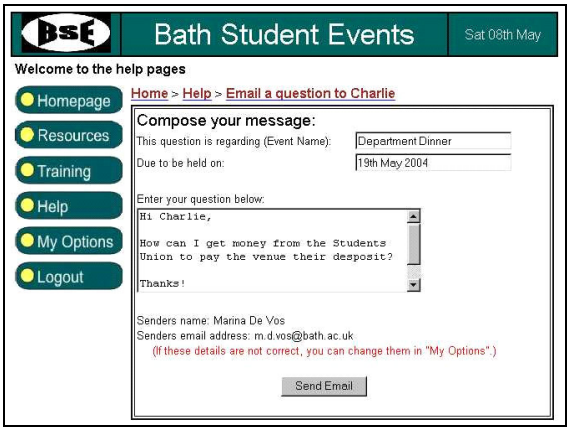


Figure 6.6 Email a question to Charile

6.6 How Implementation Differed from the Design

Some of the implemented features looked different to what was laid out in the design. There were many minor differences as would be expected between a layout created using Microsoft PowerPoint and HTML code, but two of the more significant changes are described below:

Quotes

What was initially designed for its compact and simple nature simply would not fit within the boundaries of the page. The function to “delete” or “update” a quote had also not been included in the initial layout so some changes were necessary for these to be added. Although the participants in the design focus group would have preferred the quotes to be stored on a separate page, this was decided against for the prototype to see if the original layout would work. The simpler the system is to understand, and the more appropriate the information being displayed, the better. The success of this will be discussed within the testing section.

The new layout was designed to follow the relevant design patterns, and grouped related information. For example, the company name, contact name and contact telephone number, are listed in the same column.
Figure 6.5 above shows the altered format.

Processing screen

It was originally thought that setting up each event would take a noticeable amount of time, and hence a processing screen would be required to keep the users informed of what the system was doing and why there was a noticeable delay. However, when this section was implemented, the time required to set up the event appeared instant, so the processing screen was scraped.

6.7 Summary

Although the implemented prototype does not show all the features designed for, it provides an accurate picture of the potential of the system. This allows its features to be tested with real users to establish whether or not this implementation has been a success.

7 Testing

7.1 Introduction

"Testing seeks to identify the differences between an applications expected behaviour - specified by its design, and the observed behaviour of the implemented system." [14]

Although only a prototype was implemented within the boundaries set out in the design, this testing section seeks to assess the success of its implementation.
The basis of this strategy is largely formulated around Bruegge's (2004) approach and uses well known testing techniques to form a comprehensive methodology. The theory behind the tests is explained in Appendix L Testing Theory, to allow this document to focus on the testing itself and the relevant results.

The test numbers referenced throughout this document refer to Appendix M – Test Plan and show the tests that were carried out, their expected result and if the actual result if this was different.

7.2 Testing Strategy

The testing strategy focuses on a mainly bottom-up approach, working through each stage advising corrections or alterations as seen fit. Generally, if any changes are made then previous tests must be re-run to check that their results have not been affected. Each failure must then be documented along with why the failure occurred and how it was solved, to help resolve future problems.

The testing strategy outlines 4 main sections of testing to be completed:

- Unit testing
- Integration testing
- System testing
- Usability testing

7.3 Unit testing

Breaking down the application into smaller segments to test individually makes overall testing a lot easier and simpler. Unit testing aims to rigorously test each element of the code by ensuring that every possible path through it has been executed successfully and it has passed every attempt to make it fail.

This type of testing forms a major part of the testing of the prototype, as illustrated in sections 1 to 9 of the test plan. These tests were largely successful, but there were several tests that the system did not pass. These were numbers: 1.3, 1.4, 1.7, 5.20, 6.3, 8.6 and 9.1.
A few of the larger errors are discussed here, but the complete results can be found in Appendix M.

The login page uses a small JavaScript function to check for the presence of data in the username and password fields before submitting them. If one or both of these fields is empty, the system should prompt the user to try again with a simple pop up window. This worked correctly when using the system in Internet Explorer and Netscape Navigator, but the window failed to appear whilst using other browsers tested (Opera and Mozilla Firefox).
This is not a critical error for this prototype as the user is simply remains on the login page.

[14] Bruegge, B and Dutoit, A (2004) Object-oriented Software Engineering using UML, Patterns and Java, 2nd ed., London: Pearson Prentice Hall

The other error regarding the login page was that the system accepts usernames and passwords regardless of whether they are entered in upper or lower case, when this information should be case sensitive. This is poor practice and although it is not critical to the usage of the prototype, it would not be acceptable for a working application.

In stage 1 of the event setup wizard the function that checks the validity of the event name entered but tests revealed showed that it did not allow spaces to form part of the name. This was a critical error as spaces in event names are very common. The function was altered and the other tests relating to the wizard re-run for completeness.

Another undesirable attribute of the event wizard was that regardless of whether or not the user had specified that an event date had been set, if an invalid date was entered in the drop down boxes, an error message would be presented. This signifies an issue of minor re-design because for this situation to occur, the user has to tell the system that a date of the event has not been set, and then enter a date value. (Since the default of the date input boxes depict the 1st January 2004).

7.4 Integration Testing

Unit testing seeks to check each system component individually, leading on to this next stage of integration testing to see how components work together.

The first few tests of each testing section consisted of simple visual tests to establish whether or not the page elements were working together or not. Each page was made up of several functions – the title bar, the message bar and the toolbar. The system passed all of these tests successfully.

7.5 System testing

Once the underlying code of the system is deemed to be working correctly, a higher level of testing takes place to establish the suitability of the system against a number of factors. Requirements testing and performance testing are the only parts of this relevant to the prototype, and further information can be found in Appendix L.

7.5.1 Requirements testing

The prototype met all of the basic requirements that were set out for it, such as “to provide students with guidelines on organising events”. The more subjective requirements especially regarding usability are discussed in section 7.6 below.

7.5.2 Performance testing

Some sample recovery tests were attempted (numbers 10.1 and 10.2), but these proved too difficult to test accurately, as the system performed the tested actions too quickly. This is a section which must be focussed on for any future work.

7.6 User and Usability Testing

If a system is not usable, it is worthless so this section looks to judge this. Although these sections are dictated by the requirements, they are important enough to warrant their own section and fit in well here.

Section 4.9 explains the importance of this application to meet usability standards, and refers the reader to Appendix H. This summarises some of the most well known theories on this area, and lists the resulting requirements.

Users have been involved at both the requirements and design stages of this project, with the aim of creating the most appropriate application possible. This section seeks to ascertain if their needs have been met.
This will be investigated through the medium of a user interview, whereby the assessor will fill out a questionnaire observing their performance and recording their comments.

To get a range of sample viewpoints of potential users, the two the participants who were involved in both the requirements and the design focus groups were interviewed, as well as three others who had not been involved in any part of this project so far.

The aim of the interviews was to gain an understanding of the users' overall opinions of the application, as well as measuring it against some of the usability requirements, to establish areas for improvement. Not all of the usability and user requirements were measured against, as they are not all relevant to this prototype. For example, the requirement number "User 5" states that the system must display when any changes were made to the event and by whom, for events being organised by more than one person. This is impossible to test for as event sharing has not been implemented in the prototype.

The questionnaire aimed to establish whether or not the prototype met the following groups of requirements, which are discussed along with the results below.
A copy of the completed questionnaires can be found in Appendix N.

1) System Presentation and Structure

Requirements tested: U2, U5, U6, U7, U12

Overview: The system should be easy and as logical for the user to navigate around as possible. It must also be simple to learn how to use and hence remember how to use.

How this is tested: Questions 1 to 19 observe how the user interacts with the system. Firstly, when they are guided, and then later when they are asked to complete actions unaided to see if they remember how to use it.

Results: All the users found it very easy to login and move around the system, agreeing that it was both simple and logical. After the users had been given some instructions on how to interact with the system, on returning to do the same tasks, they completed them effortlessly.
However, using the quotes section caused confusion. One participant was put off because it did not appear obvious that it was for listing company quotes; another didn't understand why the quote they had entered was not confirmed, but they hadn't checked the "confirm" box as this feature requires.
Several of the users did not understand how updating the quotes worked. They were unsure of whether they should update the details and the system would automatically save the changes, or whether they should click on the "update all quotes" button and then edit the necessary information.
The final comment from one user was that the quote management facility felt very delicate, in that it would be very easy to simple delete the quote information by mistake.

2) User Control

Requirements tested: U8, User 2

Overview: The system should make the user feel in control of their event and what they are doing, also by providing them with appropriate feedback.

How this is tested: Question 23 asks them directly if they feel that they are in control.

Results: Of the 5 participants, only 1 felt that the system was controlling him, and that he wasn't controlling it. He felt that the system was telling him what to do too much, instead of advising and encouraging.

3) Getting Help

Requirement tested: U19

Overview: The system must present helpful information to the user wherever appropriate. They should also be aware of where they can get more help should it be needed.

How this is tested: Questions 12 and 13 ask the user about the sample help facility, and additional observer comments throughout the interview pick up on any error messages that were encountered and how the user reacted to them.

Results: All of the users liked the sample help facility – to send an email to the societies administrator, but one thought that clicking the link would open their own mail facility – although the saw this implementation as a much better idea.

4) User Frustration

Requirement tested: U23

Overview: The system must avoid causing any kind of user frustration.

How this is tested: Question 25 asks them directly if there was anything that caused frustration to them whilst using the application. Comments throughout the interview should also reflect any frustrations that were encountered.

Results: Despite best efforts, the system did cause minor frustration – mainly around the quotes area as outlined above in the "General system presentation and structure" section. If these problems were eliminated then so too would the users' frustration with the system.

5) User feelings

As outlined at the beginning of Appendix H, the system should aim to be satisfying to use, motivating enjoyable etc.

How this is tested: The final section of the interview asks users to rate these feelings and if they believed the system achieved them. These are also backed up the rest of the interview as a whole, to provide an explanation for their opinions.

Results: When asked to rate their feelings, the users responded as follows:

Using the system was:	Strongly Agree	Agree	Neither agree or disagree	Disagree	Strongly Disagree
Satisfying		4	1		
Enjoyable		2	3		
Fun		2	3		
Motivating		5			
Aesthetically pleasing	1	4			
Supportive of creativity		2	3		
Rewarding		2	3		
Emotionally fulfilling			2	3	

Table 7.1 Results Of User Feelings

Whilst the participants saw the suggestion that a system could be “emotionally fulfilling” as debatable, they responded positively to the other areas in question. By the nature of this system, whilst using it is not necessarily fun and enjoyable, it supports the task of event management which is these things in the eyes of event managers. One area that could be improved is being “supportive of creativity”. However, the information and resources in this prototype are not in question within this project, so this would have to be re-tested when the system had better information inside it.

6) Additional points of interest

During these tests, many smaller points of interest were raised with regard to the systems design. Although these would require a relatively small adjustment, they can further help to decrease the amount of frustration the user feels:

- **Viewing area:** Two of the users questioned the on screen viewing area, and the layout where the content of the page is longer than the screen area. They would have preferred to have been able to scroll within a window, thus always having the title bar and toolbar in the same place.
- **Notes Button:** All of the users took longer to locate the “Click here” (to add notes) link compared to the “Add new quote” button. The three users that picked up on this requested that the notes link be replaced with a button for both ease of use and consistency.
- **Text area font:** One user commented on the change of font type within the notes section and the “extra details” in the quotes section. The rest of the application displays data using the “Arial” font type, whilst the text in these areas appeared as “Times New Roman”.
- **Look and feel:** Whilst all of the participants agreed or strongly agreed to the statement of finding the system aesthetically pleasing, one would have preferred a change in colour scheme to avoid burgundy.
- **Planning index:** The first user to complete the interview suggested that there should be an indicator on the planning index, such as a change in colour, to indicate whether or not the item had been completed or not. This was then suggested to the other participants and they too agreed. This capability was included in the design section but was not chosen for implementation. However, these comments do illustrate that this is something important for potential users.

7.7 Summary

Despite the limitations of the prototype, and the limitations of time and resources to complete satisfactory testing, this section will be able to provide a solid base of knowledge on which to improve the system – in particular the results of the user testing.
These results also highlight the advantages of prototyping, which are useful for gaining the best possible feedback of the system, and will inevitably help the final system to deliver the best possible support to its users.

8 Conclusion

8.1 Introduction

The principle objective of this project was to create an application to support students organising events by providing them with all the information they require. This final section identifies the work required to improve the application, and then seeks to establish whether or not the work completed has met its objectives.

8.2 Further Work

Many of these points requiring further action have arisen from the testing section. The prototype as it stands caused frustration to the users in both their understanding of the system, and the way they interacted with it. These simple minor alterations would delete a number of these problems, whereas the more significant problems discussed in section 8.2.2 outline the key re-design work necessary to eliminate the remaining problems.

8.2.1 Minor Alterations

1) Replace the small link that the user clicks to add their own notes, with a button proportional to that of the quotes button
2) Ensure that all elements of the system requiring data to be input from the user, display that information in the same font
3) Ask users to confirm destructive actions before proceeding.
4) Allow the planning wizard to accept the value of “other” to be entered by the user when they are asked what type of event they are planning
5) Change the title of “Quotes” to “Company Quotes”
6) Clarify the link text of “Email a question to Charlie” to show that it is not simply going to open the users’ default email client
7) Change the breadcrumb “Home” to “Homepage” for consistency between the toolbar and the breadcrumbs
8) Display events on the homepage in chronological order
9) Ensure that the users’ screen is filled appropriately no matter what screen resolution their monitor is set to display
10) In the planning index of each event, illustrate to the users which planning sections have bee completed.

8.2.2 Features requiring re-assessment

1) Quotes section

Of all the features in the system, this could be seen as the most central to the users’ work. However, it was the section that caused the most confusion with users with a design not intrinsically directing the users in its use, and the feeling that the data might be lost at any moment. It is critical to the systems success that it is both usable and trustworthy so subsequently this will require a complete re-design. This must once again, be trialled with a section of users to ensure that it meets their objectives and expectations.

2) User control

Whilst all the users showed signs of this sentiment, one user explicitly expressed his feelings that the system was controlling him – pushing him through a pre-defined route and instructing him on what to do rather than being a background information provider. Looking back to the conflicting requirement in Section 4.11.5, one part of this control

issue can be related to it – the question of how much flexibility the users are given, whilst ensuring that they organise their events appropriately.
This problem could be impractical to solve as it is never possible to please everyone. However, further design work and consultation with potential users, can help to strike a better balance.
One example of how this could be achieved is to combine the guidelines and checklist information, but this goes against Dickleman (1996) principle of a performance zone (as seen in Section 2.9.2), expressing that the optimum performance level is achieved by providing the right information *at the right time*.

Further development of this application and the re-design of those features performing unsatisfactorily, will result in a system that students can rely on to assist them in organising events to the best of their capabilities.

8.3 How the Application compares to Previous Work

8.3.1 Comparisons against the relevant Literature

One of the underlying fundamental principles running through the literature review is the power of knowledge. Used correctly and appropriately, and ever growing and developing, this knowledge can not only add more strength to an application or software system, but add to the strength of the organisation itself.

This project aimed to show how a combination of these concepts can be used to develop a strong, useful and robust application which can make a real difference to the person performing the task. That task was event management.

The performance zone defined by Dickleman (1996) in Section 2.9.2 offers a way to reach efficiency by presenting the user with enough of the right information at the right time. This application takes this one step further, taking users through distinct stages in which they are offered the relevant information. The aim was to reach that performance zone and keep the advice as relevant as possible.
However, the experienced users who took part in the testing felt far too constrained by this, and were more interested in being able to see everything that they needed to know at the same time.

Leading on from this, each of us have our own individual learning style and it is impossible to design a system perfect for all of them. However, work on developing different approaches for different types of users needs to be carried out.

The challenge of learning how to run events is however greatly lessened by the existence of this application, allowing the user to be productive from the very beginning of their event planning. Gery (1991) classifies this as an Electronic Performance Support System (EPSS) which status this project tries to reach in the most effective way.

Donating knowledge (Bush cited by Rosenberg (2001) in section 2.3) is a crucial element of the event management application. Although the storage and use of event evaluation reports was not chosen for implementation within the prototype, the design touches on how this can be achieved.

These key ideas can be seen deeply routed in the foundations of the system, influencing both its design and implementation. For any future support system to ignore these theories would be foolish.

8.3.2 Comparisons against currently available Software

Although the currently available software was designed to support event managers mainly in their task of controlling their event information, and not to tell them how to organise their events, several comparisons can be made.

Whereas it can be said with confidence that every effort was made to design this application according the users' tasks, the existing applications seem to focus on the information storage issues. Granted, the information they can store is considerable, but this may have made a hefty impact on its usability.

The event management application aimed to stay appropriate to the task – to support students and advise them of the activities they needed to carry out to organise their event successfully. Many of the reviewed packages offered many extra facilities such as inbuilt word processor. The possibility that an event manager does not have a word processing package available to them already is extremely low. Where these systems complicate matters, the event management prototype aimed for simplicity and usability, and encouraged the students to use packages they were already familiar with.

As for current advice available to students from their institutions, it is all paper based. This application pushes the barriers, both by being more appropriate, useful and more accessible.

8.4 Project Critique

A solid foundation of suitable research was crucial to the direction of this project, as well as appropriate user involvement throughout each stage. The result was a prototype that was logical and easy to understand, which the users it was tested on reacted well to. Of special interest was the result that they all considered the application to be motivational.
Because of this, the project can be deemed a success, seeing as its fundamental aim was to get students organising events effectively, and a system that motivated all of them, regardless of its drawbacks, has a future with them.

The issues with the prototype have been outlined in section 8.2 on further work required. It is important however to remember the purpose of prototyping – to reveal any errors and design flaws, on which to base further development work. So the success of this prototyping exercise forms an excellent base on which to complete further development work.

If the project was completed again, more time should be spent assessing various types of users and their needs including the completion of a full task analysis to help establish this. The research that was done would be lengthened, and many focus groups would be run to ensure that the maximum number of opinions were uncovered.
In addition, the design focus group should present several ideas and layouts to the participants, to establish an optimum application design.

For the users, the information contained in the system is the key deciding factor to its overall usefulness. Even though the content of the information did not form part of the scope of this project it would inevitably have made an impact on the users' opinions of it. To assess the true results of any further development and testing, the information should be improved.

On the more practical side of the project, the implementation could be optimised further to increase the speed of the application and maintainability of the code. With the extra redesign work that needs to be completed, it is recommended that the code is re-evaluated after the final design improvements have been established.

8.5 Concluding Remarks

To conclude, there is a clear difference in tools that provide data support, and those that aim to advise users of what actions they perform. Such an expert system is difficult to write as different users will have different ways of doing things, but there are still a set number of tasks that will be common to all of them. Despite the difficulties associated with development, the potential benefits to users are vast and will put an end to the constant “re-invention of the wheel” which is seen so widely in the Students’ Union currently.

This project has explored the theory behind the requirements for such a support system to be successful, and done a lot of work with users to establish the direction the software should take. The work outlined provides a foundation for improving the design and implementation of the current application to develop a system which can support users enabling to manage their events effectively.

9 Bibliography

Aberg, J and Shahmehri, N (2001). An empirical study of human web assistants: Implications for user support in web information systems. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, Seattle, Washington, United States, 2001, pp 404 – 411. New York: ACM Press.

Bruegge, B and Dutoit, A (2004) Object-oriented Software Engineering using UML, Patterns and Java, 2nd ed., London: Pearson Prentice Hall

Couger, D. (1996) *Creativity and Innovation in Information Systems Organi*sations. Boyd & Fraser Publishing. Co, Danvers, MA.

Dickleman, G. (1996) Gershom's Law. CBT Solutions Magazine, September/October 1996.

Gery, G.J. (1997) Granting three wishes through performance-centered design. Communications of the ACM, 40 (7) pp. 54 – 59.

Gery, G.J. (1991) *Electronic Performance Support Systems.* Tolland, Mass. Gery Performance Press.

Howell, C.C (2001) The morphing Waldo: An adaptive user interface. Performance Improvement Quarterly 40 (7).

Krueger, R and Casey, M A (2000) Focus Groups: A practical guide for applied research, 3rd ed., London: Sage.

Maylor, H (2003) Project Management, 3rd ed., London: Financial Times Prentice Hall.

Meredith, J R and Mantel, S J (2003) Project Management: A Managerial Approach, 5th ed., New York: Wiley.

Oppenheim, A N (1992) Questionnaire design, interviewing and attitude measurement, London : Continuum.

Preece, J, Rogers, Y and Sharp, H (2002) Interaction design: beyond human-computer interaction, New York: Wiley.

Raybould, B. (1995). Performance Support engineering: An emerging development methodology for enabling organisational learning. Performance Improvement Quarterly, 8 (1).

Rose C. (1999). Master It Faster. 1st ed. Accelerated Learning Institute Inc. Las Vegas.

Rosenberg (2001*) E-learning: strategies for delivering knowledge in the digital age*. London: McGraw-Hill.

Shneiderman, B. (1999). User interfaces for creativity support tools. *Proceedings of the third conference on Creativity & cognition*, Loughborough, UK, October 1999, pp 15 – 22. New York: ACM Press.

Shone, A with Parry, B (2001) *Successful Event Management – A Practical Handbook.* London: Continuum.

Soloway, E et al. (1996). Learning theory in practice: case studies of learner-centered design In: *Proceedings of the SIGCHI conference on Human factors in computing*

systems: common ground, Vancouver, British Columbia, Canada, April 1996, pp 189 – 196. New York: ACM Press.

Sommerville, I (2001) Software engineering, 6th ed., London : Addison Wesley.

Torkildsen, G. (1999) Leisure and recreation management, 4th ed., London: Spon.

Van Der Wagen, L (2001) *Event Management: For Tourism, Cultural, Business and Sporting Events.* Elsternwick, Australia: Hospitality Press Pty Ltd.

Watt, D (1998) *Event Management in Leisure and Tourism*. New York: Adison Wesley Longman Ltd.

Websites

Event management Software by Dean Evans and Associates [Online] Available at http://www.dea.com/ (accessed October 28 2003).

Online registration [Online] Available at https://www.sporg.com (accessed November 18 2003).

Definition of E-learning [Online] Available at http://www.educause.edu/ep/ep_subjects.asp (accessed December 1 2003)

Welcome to Consensus [Online] Available at http://www.consensus.co.uk/event.html (accessed November 15 2003).

GEM Features [Online] Available at http://www.2ls.com/index.php?page=gem (accessed November 14 2003).

Event and Facilities Management Software by EventPro [Online] Available at http://www.eventpro.net/ (accessed November 19 2003).

Kinetic Solutions Ltd - Conference software with a difference [Online] Available at http://www.kineticsolutions.co.uk/00000015.html (accessed November 19 2003).

Mission Business Systems and Event Master [Online] Available at http://www.event-master.com/about/index.html (accessed November 15 2003).

Event Support Guide [Online] Available at http://union.susu.org/events/eventsupportguide.pdf (accessed November 12 2003).

University of Nottingham Students' Union :: Get Involved! [Online] Available at http://www.su.nottingham.ac.uk/index.php?module=ContentExpress&func=display&ceid=612 (accessed November 12 2003).

Www.welie.com – Patterns in Interaction Design [Online] Available at http://www.welie.com/patterns/index.html (accessed April 12 2003).

Web Content Accessibility Guidelines 1.0 [Online] Available at http://www.w3.org/TR/WAI-WEBCONTENT (accessed March 1 2003).

Appendix A £200+ Event Form

The “£200+ Event Form” is filled in by students to gain permission to run future events with a budget over £200.

Appendix B Existing Software Comparison

	Consensus	GEM	Event pro planner	Kinetic	EventMaster
System					
Multi-user access		Yes	Yes		Yes?
Web-enabled	Yes	Yes			Yes
Multiple security levels		Yes	Yes		Yes
Multi-lingual		Yes			
Features					
Contact management and Call logs		Yes	Yes	Yes	Yes
Reporting - System defined	Yes	Yes	Yes	Yes	Yes
Reporting - Custom			Yes	Yes	
Action list			Yes	Yes	
Event scheduling / itinearys			Yes	Yes	Yes
Attendee management	Yes	Yes	Yes		Yes
Event budgeting			Yes		
Billing and financial management	Yes	Yes		Yes	Yes
Badge printing		Yes			Yes
Floor plan designer			Yes		
Build in word processor			Yes		
Mail merge				Yes	
Capabilities					
Event Set up wizards			Yes		
Setup event templates			Yes		
Supports previous client recall				Yes	
Supports previous event recall	Yes			Yes	Yes
Marketing Support	Yes	Yes			
Graphical views and calendars		Yes	Yes	Yes	
Exports to office products		Yes		Yes	Yes
User-definable statuses			Yes		
Manage multiple locations			Yes		
Automatic Email Change Notifications			Yes		

Appendix C Sample Report from Students' Union System

All Data

1 Location & Venue

1 A venue should be sought and booked as soon as possible. This means finalising the date of the event.

5 The Choice of venue should be considered carfully as to points raised in the rest of this

16 Confirmation should be gained in writing.

17 Confirmation of venue booking?

4 First Aid Arrangements & Medical Facilities

1 Although it may only be a small event, you will need to assess the risks and hazards to see if additional first aid provision is required

13 Are there adequate 1st Aid arrangements in place?

7 Emergency Evacuation

1 An emergency evacuation plan should be drawn up by the event organiser and where appropriate in liaison with the emergency services and the licensing authority. The ability to impliment this is essential.

2 Arrangement should be made to cover matters such as:-

3 ? Who makes key decisions

11 Have the necessary arangements been made for an emergency evacuation

11 Fire Safety

1 Fire safety must receive serious consideration.

3 If you are doing anything out of the norm of a venue, additional precautions and control should be implemented.

8 Are adequate fire prevention, detection and control precautions in place?

12 Health and Safety Issues

1 Organisers will always retain a duty of care to people working or visiting an event and safety must always be an important feature before, during and after an event.

2 The risks associated with an event / activity should be assessed beforehand and the appropriate control measures put in place. This is best completed and documented by carrying out a risk assesment.

7 Have all health & safety issues been dealt with?

24 Advertising

1 Even if you have a 10 course banquet planned, entertainment from Steps, and podium dancing Sabbaticals all perfectly planned, no-one will go unless they know that it is happening so advertise well in advance!!

2 Plan your advertising. Write a strategy outlining what you plan to do when.

Appendix D Focus Group

Chairperson's notes

Opening (Adaptation of Krueger & Casey's book (2000))

Good evening and thanks for coming – it really is very much appreciated.
The aim of my final year project is to put together a system to help students run events by offering information and resources.
You were invited because: You are all students who have organised events before / who will hopefully be running events in the future, and I want to help make that process easier for you.
We want to tap in to your experience / opinions.
There are no right or wrong answers and we expect you to have different points of view.
Please feel free to share your point even if it differs from what others have said.
I'm recording the session and taking notes because I don't want to miss anything but rest assured that no names will be included in the final report and your comments will be kept confidential.
We are just as interested in negative comments and positive ones.
Feel free to have a conversation amongst yourselves and follow up on other things people have said – don't feel that you have to respond to me all the time.
Feel free to help yourself to food and drink as we go along.

Questions for Discussion

19) Why did you get involved in organising events?

20) How many people do you usually work with on events?

21) Who are they? Friends? Other Society members? And in what capacity? A fellow organiser or an advisor?

22) Consider that you are about to start working on a new event. What are the stages you go through, starting with the event idea through to the event itself, and so on. Write your ideas on paper which will be discussed as a group shortly.

23) Have you ever come across any exceptions to this list of stages?

24) When completing a project as a group, do you set in stone when and who will complete each task? Do you use formal methods such as a Gantt chart to record this?

25) Exercise: How do you organise yourself: The set of cards in front of you are a set of tasks associated with running a sports quiz with a raffle. Imagine you have thought of these. How would you group them together / record them in a filing system? What titles would you assign to each section?

26) In front of you is a computer connected to the internet and your university file space, a telephone, local business directories, and some paper. The task you are given reflects a typical task you might complete if you were organising an event. You can use anything on this table to complete this task and record your findings. [You can do anything, call anyone, look at any website etc].

The tasks that will be used are:

- Book a transport for 50 people to see a show in Bristol
- Organise a country pub tour route
- Organise some decorations for a dinner in the Claverton Rooms

- Book a restaurant for a arts show meal for 35 people
- Book a balloon modeller to appear at a department ball

27) What skills do you think you need to be a successful event manager?

28) What support do you currently receive? Would you like to have more support?

29) Consider that a system was designed to support student event managers like you, -rate these functionalities / features in order of importance.
[Cross out lightly the features that you would never use].

30) If summaries and organiser evaluation/hints/tips of previous events were available, would you use them?

31) If asked to, would you complete a short evaluation of your event for the benefit of others? [Also, event summary]

32) All things considered, would you be seriously interested in using such a computer based system? Or simply as a resource facility?

33) Do you think that a system would be beneficial to students who had never organised an event before?

34) Do you have anything else you would like to add about organising events, or a potential events support system?

35) Would you be interested in commenting on the design of this system?

36) When the system is finished, would you be interested in testing it?

Activity

Contact Backstage technical services

Discuss requirements and a price

Arrange transfer of funds

Book Audio Visual equipment

Choose entertainment

Confirm entertainment bookings

Check entertainment's credentials – Public Liability insurance etc

Get Quiz Winners Trophy

Design Tickets

Print tickets

Design posters

Print posters

Design bathstudent.com advert

Ensure that bathstudent.com advert is implemented

Email sports groups advertising the quiz

Write advert for the team bath web site

Book library foyer for ticket selling

Put up posters

Sell tickets

Complete the £200+ events form

Complete finance control sheets

Complete cash flow predictions

Create signage

Put up signage

Get sponsorship

Get prizes

Buy raffle tickets

Write a Risk Assessment

Have Risk Assessment checked

Write the Quiz

Write thank you letter to prize providers

Transcript

This is an extract from the transcript of the Focus Group.

BS I'd put that on its own

SL If Id have companies, id probably have thank you letters in advance, it be "if they'd donated a prize', id have collected it and, unless it was a charity, then id have collected it and kept it to let them know how much it had Raised or something

RA so you need double headers, you could do the letters template

SL mine was more of what kind of areas can you put together, and how you can get them to people

SJ Sponsorship and prizes could come under the same thing

SL id put sponsorship in that one

SJ nah, you see id put those two as two separate things

SJ I'd sort of use them simultaneously

SL maybe you could have kind of stared things

SJ that's more of prizes

SL that could be targeted as one main sponsor and then maybe smaller pries

SJ where as we had…

SL you need a Sarah Cooper

SL I'd put like I'd stared something, that id put like you'd probably have somebody useful, not maybe the person that was in charge, cause I did it by splitting things out, then reconvene to see how they are getting on with progress

CR so the thing was, if, I'm trying to turn this more now into if you had a system that was doing this for you, how would it kind of effect you, cause essentially, say it might split sponsorship and prizes, how would you find it, would you find it a bit bizarre or would you like to dictate what it was doing as well

BS I think if you could dictate as well

SL I cant think of any one right or wrong way, the fear is that it would churn out the way, it would tell you to do this, this and this, and maybe it would split the prizes, and you might find it, it is so individual the way you do it, you might find that you prefer to do it in a different order, or you would prefer to do something Slightly differently

BS just everyone write a to do list, the write the numbers against it, then Cross one thing off, and re write all the numbers, completely differently to the way it was first

SL I don't tend to write number on my to do lists, I look at it a go what can I actually do now, I'm sat at home, I cant do computers, I can do ….., what can I do from home now? What can I do on the computer now, who can I ring now? And then I'd Cross things off and I tend to keep that list even with things Crossed off, and then I know.

BS I tend to number it as to what is most important, but I don't necessarily Cross things off in that order, does that make sense?

SJ I tend to have big to do lists

SL if I have lots of things going on at the same time, like last week with course work, bierkeller, international type stuff, then I got to re-type that stuff

RA I wish I had your life, three things to do. That would be great

SL but it was like I had to have 4 heading and things then in side those 4 things.

CR now the object of this is to try and see how your mind works, how you would do a task, now Id like to imagine that you are sat in front of a computer, connected to your file store, so you have access to all the resources that you need, you've got a telephone, whatever, all your numbers, all your previous contacts, and I want you to describe to me how you would think about going about a task, I've got 5 different ones, and id like you to start off by giving me your own ideas, rich we will star with you, then we will discuss it and see if people want to add anything

CR if you were booking transport for 50 people to go and see a show in Bristol, what would your immediate reaction to this be, would you go to the Internet, would you pick up the phone, what would you do?

RA what do I already know?

CR you know that you are going to see a show, you know the date, you know all the details about that

RA internet yellow pages, that sort of thing, find out transport companies, telephone, call them, email, depending how technologically advanced, there probably not, hey if some one else is paying the phone bill, just phone them, it will be quicker to get a response anyway

CR so the information you got, what would you do with it?

RA if I was well behaved, id put it in a spreadsheet, and if I was at work I would put it in a spreadsheet

CR ok, what are you likely to do

RA put it in a spreadsheet, sad puppy I know

CR Would anyone else approach it differently?

BS I walk across the office, pick up the piece of paper will a list of all the coach companies and details on, pick the one with the latest cancellation for the least money, then ring them up and see if they have got any coaches, or I would ring pat.

SL if you'd have someone before, probably ring them to get a bench mark idea as well, so you know how much you are looking like paying

CR ok, next task, Ben. If you are organising a country pub tour

BS What do I do? Open the M drive, go soc secretary, pub Crawl route 1998 – 2003. Then take out the pub that does not like us anymore, and ring the map, or go for the route that I have been eyeing up for the last 3 months.
Ring said pubs on route, probably write them down on a bit of paper, cause I'm a bit antiquated like that, then I think I would think, oh I should be a bit more organised, and put them in a spreadsheet, then id go onto multimap, and type in said postcodes from Yell.co.uk, print the out, one each page, staple them together, and give them to the driver that will probably loose them in the first pub. Id also go to a meeting and find out how many people wanted to go, or I might email, and I would remember to charge the b@@@@@@@s before they go, Rather than trying to get the money back after the trip.

CR would anyone else do anything in a different way

RA yes, on my various flying jaunts, I find out pubs that id like to go to, and go round in like a car to check it out.

SL yeah do like a pre-pub Crawl

CR sal, organising decorations, for dinner in the Claverton rooms,

SJ it is a posh thing or casual

CR posh

SJ phone up various people, look in the yellow pages and see who you find for helium or gas balloons,

RA phone a florist

CR any one else

SL I'd buy potatoes and tin foil to make weights

CR I did tell the new freshers week organisers they were not allowed to use potatoes, in case they got thrown at the union president!

BS I'd find a driver to go and get said things.

CR ok, Jo, book a restaurant for an arts show meal, 35 people, how up your street is that?

JC see who is going, and see were people want to go

RA you're the organiser, you can decide were you want to go

JC no, get a general consensus, I wouldn't ask all of them, maybe use a restaurant that we have used, that we know can cater for that many, arrange a date, phone the restaurant,

CR how would you get the phone number

JC yellow pages

CR SLou, book a balloon modular for after dinner at a department ball

SL I'd probably just actually look up the people that we have had before, because they were very good at it, I may try and look out someone else and compare the price, and check that we were not being ripped off, but I would not look them up on the internet, cause I'm not a fan of yell.com
It would probably be written in a ball diary of somewhere, id give them a ring and see if the were available, and if they weren't if they could recommend anyone else. If I had to look up a number, id use the yellow pages, I think there is a section, but like a lot of other companies may have contacts, party hire as well, may have people they use

BS suppose the other place you could go would be an events firm, but then you would be paying extra.

CR functionality, I know that I think you have looked at these on the web, but they are useful to discuss, would like to look at what would be good for a system and what wouldn't be List of ideas

BS I don't know about a list of ideas, but a list of previous events, not sure that listing all the ideas that somebody has ever had would necessarily be useful.

SL sometimes ideas are really good, just things that have never been done, like something that you would just never necessarily have thought of, like I found a really good web site the other day, it was linked from the girl guiding web site, and it is just a list of events, and a list of themes, so say you wanted to do Alice in wonderland, there is a list of all the things you could do, and it is not necessarily that they have done it, but an idea that someone might want to do.

BS yes, but is the danger that if it is just a list, that some of them are just bad and wrong?

SL but at some point the ball was just an idea

BS yes, but what I'm trying to say is that nicking the VC's car, and dropping it off a Crane or something

SL but that is never going to be on a list of ideas

BS but you don't know that if it is just a list of possible ideas, cause Rag, I mean national Rag, used to do events that are now totally not acceptable

SL there is that element, but it is a matter of stating that, and there would be someone administrating the system and would be obvious to them that things like dropping the VC's car is not, and you would think that they would have the sense not to put that on in the first place. Things like a Slave auction came from a rower, it was a new idea that could have been got from an idea list

CR what about from other universities or related type groups?

BS depends on how much work you want to make for the 'censor'

CR right, things you need to know e.g. if you said we are doing an event for 1000, its going to say 'safety', you need to do a risk assessment and get it checked by Geoff, blah, blah bla, is that useful?

SJ for people that it is there first time round, they don't know things like that

BS you need a list of things you need to know, definitely, but then we are talking stupid people, or people that do not understand the concept of loosing money, or for example the latest Rag project, seeing how many people you can fit in a mini, Geoff did not know until he had a phone call from St john, there almost needs to be some thing in the system that says oi, I'm not going any further till you have spoken to Geoff, which is what would happen if it wasn't a computerised system

Appendix E Questionnaire Examples

Experienced Event Organiser Questionnaire

There are no right or wrong answers; I merely want an idea of what and how you do things.

If you have never been the organiser of an event, please go to the novice questionnaire and fill out that survey instead.

Your answers will be kept confidential.

1) How many events have you organised or helped to organise? This can include events before you came to university such as a school ball?

- ☐ 0
- ☐ 1-2
- ☐ 3-5
- ☐ 6+

2) What were the reasons that you got involved in organising events? Tick all that apply.

- Friends doing it ☐
- Wanted to raise money for charity ☐
- Someone had to do it and no-one else volunteered ☐
- Part of my job as club/society/school officer ☐
- Wanted something extra to put on my CV ☐
- Just fancied having a go at something different ☐
- Didn't have a particular reason ☐
- Other ☐

3) Briefly name and describe the 1 most significant event for you, that you have helped organise. This may be the biggest, or the more recent.

Name ☐

Brief description ☐

E.g. department social last week with 30 people

With this event in mind, please answer the following questions:

Support

4) What support did you receive when planning the event? Tick all that apply.

None ☐

Friends and family ☐

School staff ☐

Union staff ☐

University Staff ☐

Other _____

5) Would you have preferred to have more support available to you?

○ Yes

○ No

○ Don't Know

6) If yes, briefly describe what further support you would have appreciated.

_____

Skills

7) What skills do you think you need to be a successful event manager? Put all that apply in order of importance. 1 = most important.

Time management _____

Common sense _____

Organised _____

Calm under pressure _____

Problem solving _____

Flexible _____

Committed _____

Other1 _____ _____

Other2 _____ _____

Other3 _____ _____

Organising yourself

8) When you started to gather information on different tasks, e.g. quotes for transport, cost of different venues, what do you use to keep track of all the information? (Tick all that apply)

- ☐ I don't use anything
- ☐ PDA
- ☐ Personal Computer - e.g. with a spreadsheet or database
- ☐ The back of a notebook
- ☐ A separate notebook bought for the task
- ☐ Paper stored and ordered in a file
- ☐ Loose sheets of paper kept in various locations
- Other ☐

9) When you work with other organisers on an event, how do you divide up the tasks and keep track of time? (Tick all that apply)

- ☐ I have never worked in a group
- ☐ Use a gantt chart
- ☐ Timetable tasks formally
- ☐ Assign tasks to people as we go along
- ☐ Don't know
- Other ☐

System

10) If an online system was available to help you organise events, which of these features would you find useful? Rate all that apply in order of importance (1 = highest)

- Lists of event ideas ☐
- List of things you need to know - for example, suppliers have public liability insurance ☐
- List of resources - e.g. potential venues, recommended coach companies ☐
- Help with budgeting ☐
- Help with filling out union forms ☐

Ability to manage quotes from suppliers ☐

Lists of event evaluations by other organisers ☐

Summaries of previous events - showing where / when / what suppliers were used etc ☐

Checklists to help you plan ☐

Sample events for learning ☐

Fun games and quizzes to help build key skills - (as listed in question 7) ☐

11) If summaries and organiser evaluation/hints/tips of previous events were available, would you use them?

- ☐ Yes
- ☐ No
- ☐ Don't Know

12) If asked to, would you complete a short evaluation on your event for the benefit of others?

- ☐ Yes
- ☐ Yes - if there was an incentive
- ☐ No
- ☐ Don't Know

Overall

13) All things considered, would you be seriously interested in using such a computer based system or would you rather carry on as you are.

- ☐ Yes - I would be interested
- ☐ No - I like the way I do things
- ☐ No
- ☐ Don't Know

14) How about as just a resource facility?

- ☐ Yes
- ☐ No
- ☐ Depends
- ☐ Don't Know
- ☐ Other ☐

15) Do you have any further comments? ☐

The next stage

16) When the system is finished, would you be interested in testing it?

- ☐ Yes
- ☐ No
- ☐ Don't Know

17) If yes, please fill the following details:

Name

Bath email address

Thank you very much for taking the time to complete this survey - it is greatly appreciated.

Send Reset

Novice Event Organiser Questionnaire

There are no right or wrong answers; I merely want an idea of what and how you do things.

If your have experience in organising event, please go to the experienced questionnaire and fill out that survey instead.

Your answers will be kept confidential.

1) How many events have you helped out with, as a volunteer /marshal or paid worker?

- ☐ 0
- ☐ 1-2
- ☐ 3-5
- ☐ 6+

2) What were the reasons that you got involved in helping out at events? Tick all that apply.

- Friends doing it ☐
- Wanted to raise money for charity ☐
- Someone had to do it and no-one else volunteered ☐

Part of my job as club/society/school officer ☐

Wanted something extra to put on my CV ☐

Just fancied having a go at something different ☐

Didn't have a particular reason ☐

Other

Skills

3) What skills do you think you need to be a successful event manager? Put all that apply in order of importance. 1 = most important.

Time management

Common sense

Well organised

Calm under pressure

Problem solving

Flexible

Committed

Other1

Other2

Other3

4) If you were to ever organise an event yourself, would you like help in developing these skills with regards to event management, by way of advice or an informal quiz/game?

- Yes
- No
- Don't Know

Organising yourself

5) When you gather information, e.g. coursework literature, what do you use to keep track of all the information? (Tick all that apply)

I don't use anything ☐

PDA ☐

Personal Computer - e.g. with a spreadsheet or database ☐

The back of a notebook ☐

A separate notebook bought for the task ☐

Paper stored and ordered in a file ☐

Loose sheets of paper kept in various locations ☐

Other

6) When you work with others in a group e.g. on a coursework project, how do you divide up the tasks and keep track of time? (Tick all that apply)

I have never worked in a group ☐

Use a gantt chart ☐

Timetable tasks formally ☐

Assign tasks to people as we go along ☐

Don't know ☐

Other

System

7) If an online system was available to help you organise events, which of these features would you find useful? Rate all that apply in order of importance (1 = highest)

Lists of event ideas

List of things you need to know - for example, that it is recommended that suppliers(e.g. Photographer, karaoke…) have public liability insurance

List of resources - e.g. potential venues, recommended coach companies

Help with budgeting

Help with filling out union forms

System to help with managing quotes from suppliers

Lists of evaluations of events by other organisers

Summaries of previous events - showing where / when / what suppliers were used etc

Checklists to help you plan

Sample events for learning

Fun games and quizzes to help build key skills - (as listed in question 7) ☐

8) If summaries and organiser evaluation/hints/tips of previous events were available, would you use them?

- ☐ Yes
- ☐ No
- ☐ Don't Know

9) If asked to, would you complete a short evaluation on your event for the benefit of others?

- ☐ Yes
- ☐ Yes - if there was an incentive
- ☐ No
- ☐ Don't Know

10) Do you have any further comments? _____

The next stage

11) When the system is finished, would you be interested in testing it?

- ☐ Yes
- ☐ No
- ☐ Don't No

12) If yes, please fill the following details:

Name _____

Bath email address _____

Thank you very much for taking the time to complete this survey - it is greatly appreciated.

Send Reset

Appendix F Questionnaire Results

Experienced Questionnaire Results

How many events have you organised or helped to organise? This can include events before you came to university such as a school ball?

1-2 Events	11
3-5 Events	12
6+ Events	11

What support did you receive when planning the event? Tick all that apply.

None	2
Friends and Family	21
School staff	2
Union staff	20
University staff	19

CR: lecturers, people involved - e.g. safety office, VC etc!

Other responses given: Course mates
More experienced group members (e.g. rag, society)
Event staff
External company - x2

Would you have preferred to have more support available to you?

Yes	9
No	19
Don't know	6

More support:
- Comprehensive hand-over reports
- Better understanding from volunteers
- Clearer div of responsibilities of University staff
- More people

CR: Suggests training areas…

When you started to gather information on different tasks, e.g. quotes for transport, cost of different venues, what do you use to keep track of all the information? (Tick all that apply)

Nothing	0
PDA	2
PC	9
Random paper	5
Spec. notebook	19
Papers in file	18
Loose sheets (mess)	8
Other:	0

Comment: Random paper result to do with not have a PC avail all the time?

When you work with other organisers on an event, how do you divide up the tasks and keep track of time? (Tick all that apply)

Never have	0
Gantt Chart	7
Formal timetable	13
Assign on the way	26

Don't know	1

Other responses: Excel sheet
Verbally

Comment: therefore formal scheduling would be useless
Comment: Can see start of intuitive use of computers

If an online system was available to help you organise events, which of these features would you find useful? Rate all that apply in order of importance (1 = highest) Potential features

	Average score	Rank
Guidelines	3.10	1
Previous event summaries	3.23	2
Resources	3.24	3
Checklist	4.10	4
Budget help	5.34	5
Event eval	5.82	6
Ideas	5.89	7
Manage quotes	6.50	8
Sample events (learn)	7.39	9
Form help	7.84	10
Training fun	9.91	11

If summaries and organiser evaluation/hints/tips of previous events were available, would you use them?

Yes	32
No	0
Don't know	1

If asked to, would you complete a short evaluation on your event for the benefit of others?

Yes	23
Yes if incentive	1
No	3
Don't know	6

All things considered, would you be seriously interested in using such a computer based system or would you rather carry on as you are.

Yes	27
No	2
Don't know	4

How about as just a resource facility?

Yes	26
No	0
Depends	5
Don't know	1
Other	0

Do you have any further comments?

Using the system depends on robustness, and design
Reading previous event reports depends on how relevant they are to my event.
Ordering by relevance is good.
A "Getting started" supplement is useful
Would use as long as don't have to do masses of data entry to get going!

Novice Questionnaire Results

How many events have you helped out with, as a volunteer /marshal or paid worker?

1-2 Events	6
3-5 Events	1
6+ Events	2

When you gather information, e.g. coursework literature, what do you use to keep track of all the information? (Tick all that apply)

Nothing	0
PDA	0
PC	3
Random paper	2
Spec. notebook	3
Papers in file	3
Loose sheets (mess)	5

Other: A number of random docs on the computer

When you work with others in a group e.g. on a coursework project, how do you divide up the tasks and keep track of time? (Tick all that apply)

Never have	0
Gantt Chart	1
Formal timetable	2
Assign on the way	9
Don't know	0

If an online system was available to help you organise events, which of these features would you find useful? Rate all that apply in order of importance (1 = highest)

	Average Score	Rank
Resources	1.57	1
Guidelines	2.89	2
Previous event summaries	3.63	3
Budget help	4.29	4
Checklist	4.33	5
Event evaluations	5.00	6
Manage quotes	5.67	7
Ideas	6.50	8
Form help	7.00	9
Sample events (learn)	7.50	10
Training fun	9.17	11

If summaries and organiser evaluation/hints/tips of previous events were available, would you use them?

Yes	8
No	0
Don't know	1

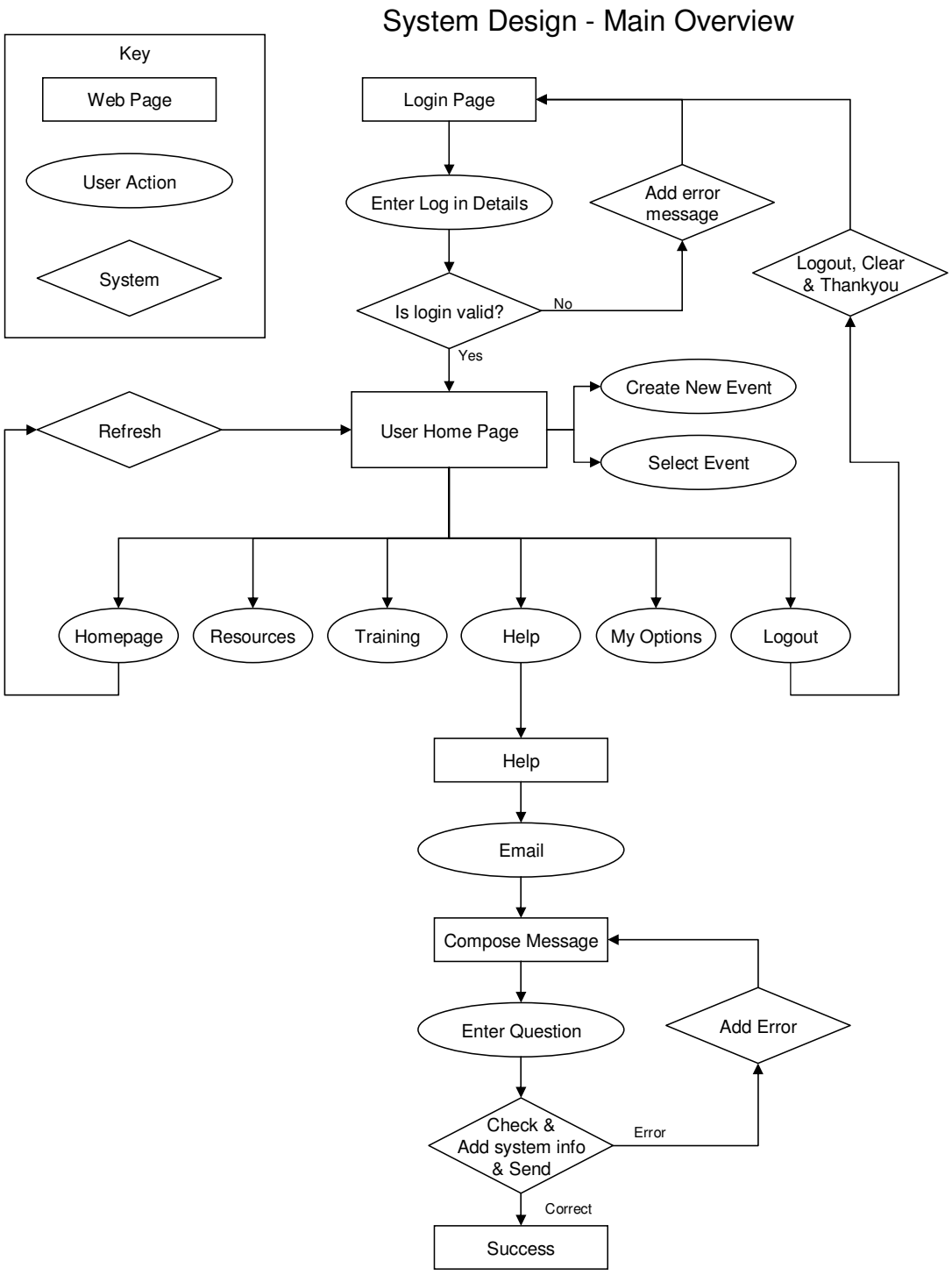
If asked to, would you complete a short evaluation on your event for the benefit of others?

Yes	7
Yes if incentive	1
No	0
Don't know	1

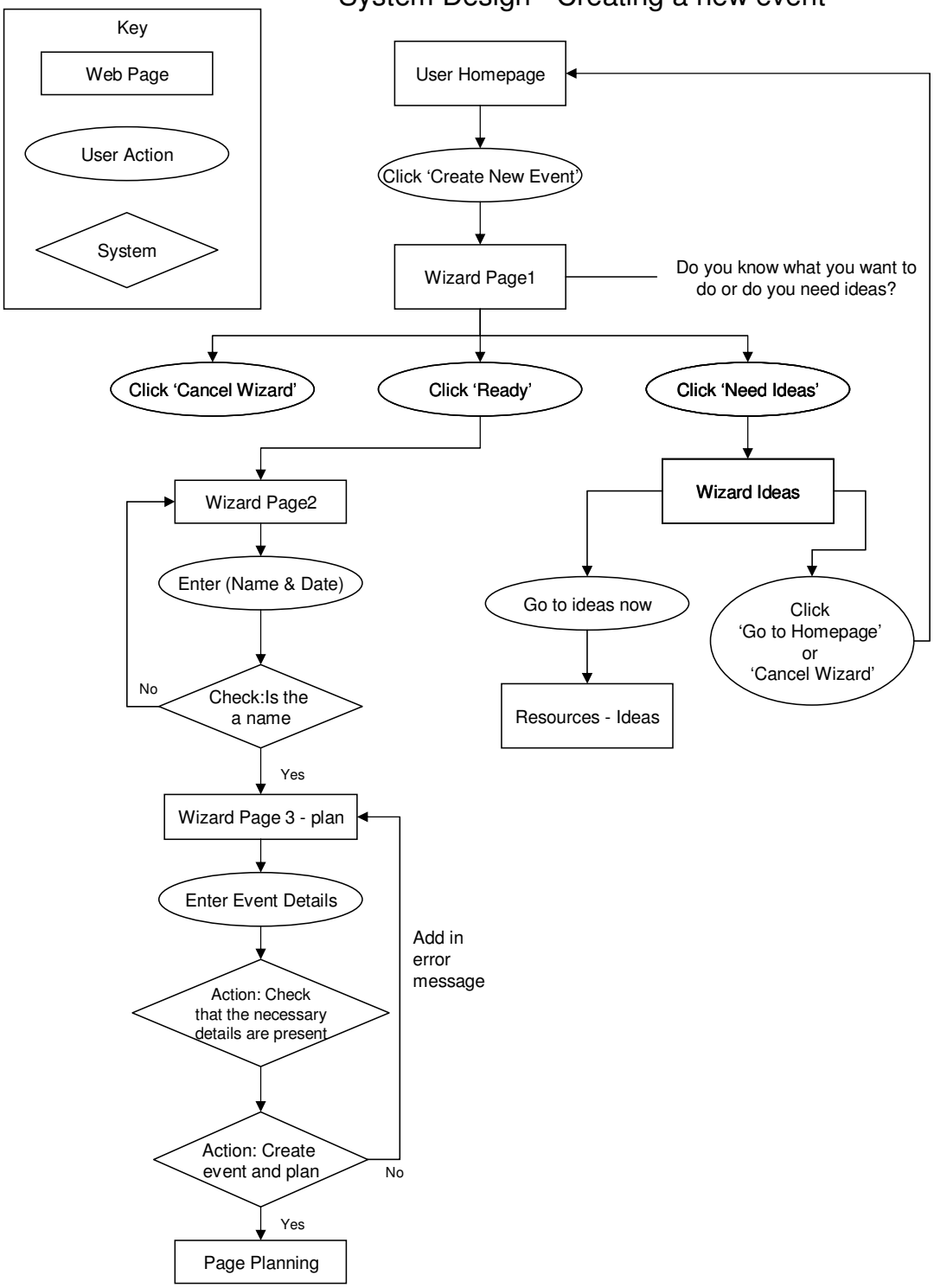
Do you have any further comments?

None

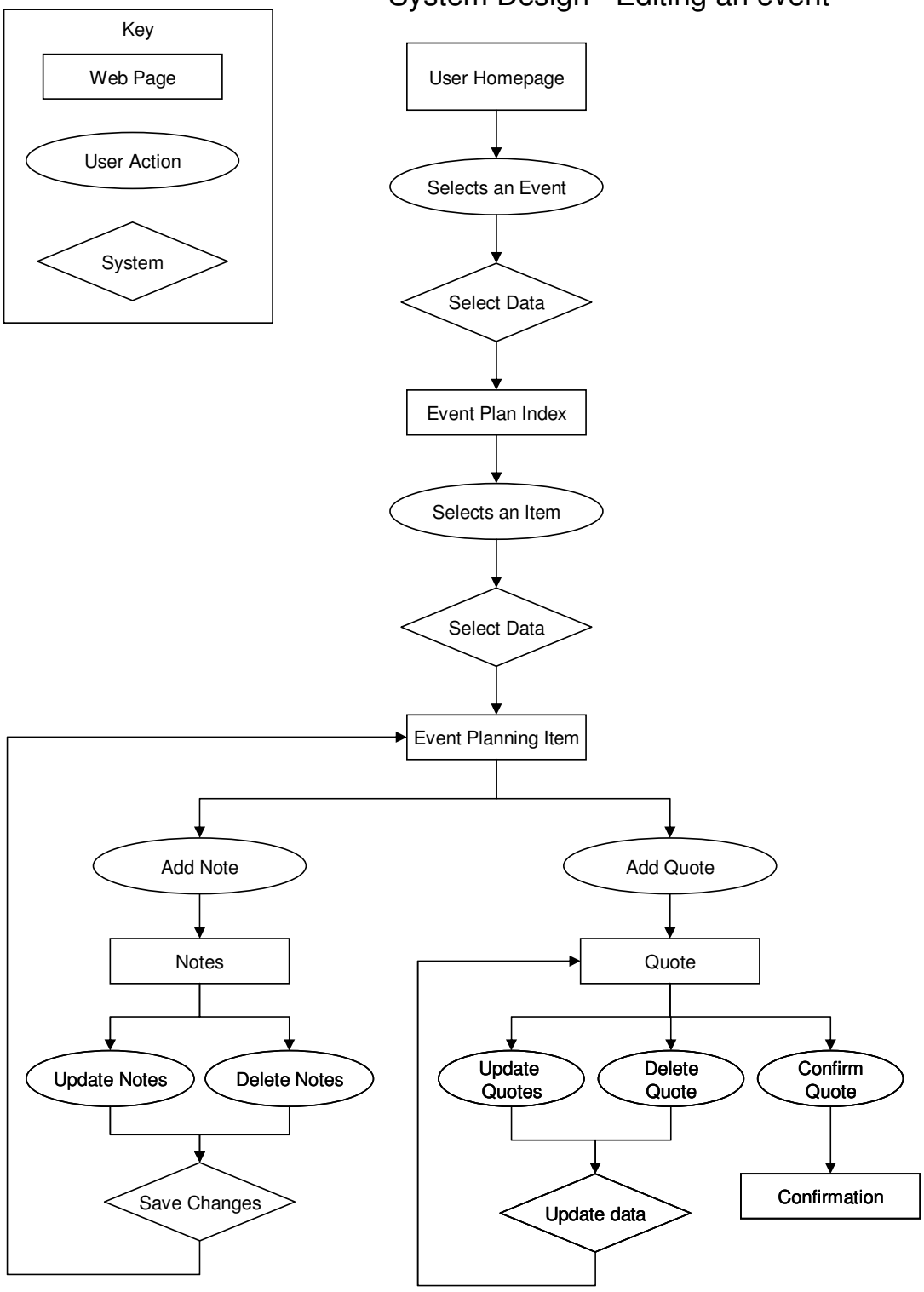
Appendix G System Flow Diagrams



System Design - Creating a new event



System Design - Editing an event



Appendix H Usability Requirements

Preece (2002) outlines some of the key theories on designing 'usable' applications, which are summarised here with their related resulting requirements. The aim is to provide a pleasant experience for the user by designing a system that feels nice to use. By meeting the requirements outlined below, an application aims to create a system that is:

▪ Satisfying		▪ Motivating
▪ Enjoyable		▪ Aesthetically pleasing
▪ Fun		▪ Supportive of creativity
▪ Entertaining		▪ Rewarding
▪ Helpful		▪ Emotionally fulfilling

Please note that this document lifts much of the material directly from the texts referenced.

Design Principles

1. Applications should be good at doing what they are supposed to do. This is a very general goal, and asks if the system is capable of allowing users to achieve their objectives using the system. For example, helping them learn, carry out their work effectively and so on.

Requirement U1: The system should allow users to carry out their tasks as defined in the use cases in Section 4.8.

2. An efficient system allows users to complete common tasks through a minimum number of steps. E-commerce sites are a good example of this, requiring users to enter their details only once and allowing them to make purchases on many occasions.

Requirement U2: Once the users have learnt to use the system, they should be able to sustain a high level of productivity using it.

3. A system is deemed "safe" if it protects the user from undesirable situations. Examples of this are asking users to confirm destructive actions such as deleting information, and also to allow them to undo their actions if they so wish.

Requirement U3: The system should prevent users from making too many serious errors, and allow them to recover from them when they do occur.

4. A system is said to have good 'utility' if it provides users with the right functionality to allow them to achieve their objectives. These functions must not restrict the user though, such as a graphics package may do by not allowing users to draw free-hand.

Requirement U4: Users should have the freedom to complete their tasks in the way they want, by being provided with an appropriate set of functions.

5. Systems should require only a very low level of training (if any) until they can be used and understood by a new user. Preece (2000) cites Nelson's "10 minute rule" proposing that where appropriate, a novice user should be able to learn and understand a new system in under 10 minutes. Taking advantage of users existing knowledge of computer systems can help achieve this. It would not however be appropriate in the case of a pilot learning to fly a plane for example.

Requirement U5: Users must be able to learn how to use the system in under 10 minutes.

6. Once a user has learnt to use a system, it should be easy for them to remember how to do it. This is especially important for systems which are used infrequently. The systems design is crucial to this, and should use meaningful graphics, contain handy reminders and have a good structure.

Requirement U6: It should be easy for users to remember how to use the system by being logical and well structured with good interface support.

Preece (2002) then goes on to cite Norman's (1998) design principles:

1. Visibility. The more visible the systems functions and features are, the more likely the user will be to find them and know how to use them.

Requirement U7: The system must present its functions to the user in an obvious way, such that they will able to easily anticipant how to interact with them.

2. Feedback. Related the previous concept, appropriate feedback can provide the necessary visibility for human interaction. Feedback allows the user to understand how they are progressing through a task, giving them information to allow them to continue with it.

Requirement U8: The user must always be able to understand what point they are at in a task, by receiving appropriate system feedback.

3. Constraints. A system can help users prevent getting themselves into erroneous states by constraining their actions. A common example is applications that "grey out" items on menus if selecting them would be inappropriate. Other constraints that can be applied are logical ones, that rely on people's common sense and reasoning about actions and their consequences. Cultural constraints rely on already learned conventions, such as the use of red for warning signs.

Requirement U9 (Similar to requirement U3): The system must constrain the user from entering erroneous states and seek to present possible actions logically. For example, the colour red can be used to present warning and error messages.

4. Mapping. This refers to the relationship between controls and their effects in the world. A good example is the layout of cursor keys on a keyboard, or how a remote control is laid out. The system should map its features in a logical way to avoid confusing users or forcing them to adapt to a new layout.

Requirement: U10: The system must be laid out in a logical order than draws on users' previous experience of using artefacts (such as a remote control) where possible.

5. Consistency. A consistent application is one which use similar elements for achieving similar tasks, eg. Click the left mouse button to select an item.

Requirement U11: The system must have a consistent approach.

6. Affordance. This is the term that refers to an attribute of an object that allows people to know how to use it. For example, a mouse button invites clicking. In an application, the design of elements such as graphics and buttons can assist users in knowing how to use them. Tip text and labels are common examples.

Requirement U12: The system must show instruct users how to interact with it as logically and effectively as possible.

Usability Principles

Preece (2002) continues on to cite Nielsen's usability principals. Some of them overlap with the design principals discussed above but are also listed here for completeness:

1. The user should always be kept informed of what stage they are in a sequence of tasks, with a system providing appropriate feedback.

Requirement U13: The user should always be aware of what the application's current status is.

2. The system must match tasks in the real world to those in the systems as naturally as possible. One way that this can be achieved is through the use of language which must be appropriate to the user, and not orientated around system buzz words.

Requirement U14: Users must never be confused by either the layout of the system or the language used to present it's information.

3. The user must always feel in control of the system, and not restrained by its structure. A way for the user to leave a screen or area of the system and return to a state they are familiar with must always be clearly visible.

Requirement U15: The system must cause the user to feel in control of the system and what it is doing at all times.

4. A consistent system is one that always maintains set standards and does not behave erratically. It avoids puzzling users – for example, by avoiding using the same words to mean different things.

Requirement U16 (Similar to requirement 11): The system must maintain consistency to avoid confusing users.

5. Errors are generally states that the user is not expecting, so a useful system will help them recognise, diagnose, and recover from errors by presenting meaningful error messages and by describing potential solutions.

Requirement U17: The system must help users to understand why an error has occurred, and provide a way for the user to put the problem right quickly, then allow them to progress.

6. Whilst some errors are inevitable, a good system will prevent as many as possible from happening in the first place.

Requirement U18: The system must prevent as many erroneous states as possible.

7. A good system will make objects, actions and options visible to the user. They should not have to remember information from one part of an application to another and instructions for how to use the system should be obvious or easily retrieved where appropriate.

Requirement U19: The system must present helpful information to the user wherever appropriate. They should also be aware of where they can get more help should it be needed.

8. A system should cater for experienced as well as novice users. Accelerators should be provided that are invisible to novices, but allow experts to complete a task quicker. One example is the provision of shortcuts keys and macros in Microsoft Office that can be designed by individual users.

Requirement U20: Accelerators should be provided for expert users to speed up their use of the system, but which do not distract novices.

9. Systems should avoid irrelevant or rarely needed information. Every extra piece of information presented competes for the users attention which those that are relevant – weakening their impact.

Requirement U21: The system should provide a minimal, aesthetically pleasing design.

10. Clear help and documentation should be provided by systems to assist their users in a meaningful way. It should include a clear help facility and provide step by step guides for completing key tasks.

Requirement U22: Help must be provided in a clear useful fashion, allowing the user access to further information if and when they have the need for it.

User Frustration

Frustrating users whilst they use an application must be avoided at all costs. At best, the user may leave their desk for a breath of fresh air or to make a cup of coffee before returning to it, and at worst, extensive and prolonged frustration can cause complete disregard of a system.

There are many reasons why users feel frustration and Preece (2002) lists the key high level sections of them:

- When an application doesn't work properly or crashes
- When a system doesn't do what the user wants it to do
- When a user's expectations are not met
- When a system does not provide sufficient information to enable the user to know what to do
- When error messages pop up that are vague, obtuse or condemning
- When the appearance of an interface is garish, noisy, gimmicky or patronising
- When a system requires users to carry out many steps to perform a task, only to discover a mistake was made somewhere along the line and they have to start all over again

Requirement U23: The system must avoid causing any kind of user frustration.

Appendix I Accessibility Requirements

The World Wide Web Consortium provides an excellent resource outlining key guidelines to make websites accessible for all user groups, as well as providing many real life examples and ways they can be implemented.

These guidelines are summarised below, with a resulting requirement summarising it.

Please note that this document lifts much of the material directly from the web consortiums website which has been inserted and summarised here.

1) Provide equivalent alternatives to auditory and visual content

This emphasises the importance of providing text equivalents of non-text content (such as images, pre-recorded audio, video). Text can be readily output to speech synthesizers and braille displays, and can be presented visually (in a variety of sizes) on computer displays and paper.

Coversely, websites can also supply audio and visual aids as an equivalent to text, to benefit non-readers.

Non text content that should have an equivalent to it include:
Images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ascii art, frames, scripts, images used as list bullets, spacers, graphical buttons, sounds (played with or without user interaction), stand-alone audio files, audio tracks of video, and video.

Requirement A1: Where non-textual content is used, provide appropriate alternatives.

2) Don't rely on colour alone

Ensure that text and graphics are understandable when viewed without color as many people have colour deficits, such as colour blindness.

How to achieve this:

- Ensure that all information conveyed with color is also available without color, for example from context or markup.
- Ensure that foreground and background color combinations provide sufficient contrast when viewed by someone having color deficits or when viewed on a black and white screen.

Requirement A2: The system must not rely on colour as its only form of convey a certain piece of information.

3) Use markup and style sheets and do so properly

Mark up documents with the proper structural elements. Control presentation with style sheets rather than with presentation elements and attributes.

How to achieve this:

- When an appropriate markup language exists, use markup rather than images to convey information
- Use style sheets to control layout and presentation
- Use relative rather than absolute units in markup language attribute values and style sheet property values
- Use header elements to convey document structure and use them according to specification.

- For example, in HTML, use H2 to indicate a subsection of H1. Do not use headers for font effects.
- Mark up page elements properly – e.g. lists and list items, blockquotes (in HTML).

Requirement A3: The system should use markup and style sheets properly.

4) Clarify natural language usage

Use markup that facilitates pronunciation or interpretation of abbreviated or foreign text.

How to achieve this:
- Provide expansions of abbreviations and acronyms
- When content developers mark up natural language changes in a document, speech synthesizers and braille devices can automatically switch to the new language, making the document more accessible to multilingual users.
- Natural language markup also improves readability of the Web for all people, including those with learning disabilities, cognitive disabilities, or people who are deaf.

Requirement A4: The system must provide explanations on pronounciation of abbreviated or foreign text.

5) Create tables that transform gracefully

Ensure that tables have necessary markup to be transformed by accessible browsers and other user agents.
Tables should be used to mark up truly tabular information ("data tables").
Content developers should avoid using them to lay out pages ("layout tables").

How to achieve this:
- Group tables into sections
- Provide abbreviations
- Do not use tables for layout unless the table makes sense when linearised. Otherwise, if the table does not make sense, provide an alternative
- Provide summaries for tables

Requirement A5: Where tables are used, they must still be meaningful for users when linearised.

6) Ensure that pages featuring new technologies transform gracefully

Ensure that pages are accessible even when newer technologies are not supported or are turned off.
Although content developers are encouraged to use new technologies that solve problems raised by existing technologies, they should know how to make their pages still work with older browsers and people who choose to turn off features.

How to achieve this:
- Organize documents so they may be read without style sheets.
- Ensure that equivalents for dynamic content are updated when the dynamic content changes.
- Ensure that pages are usable when scripts, applets, or other programmatic objects are turned off or not supported. E.g. the “NOSCRIPT” tag in HTML.
- Careful with Frames. Use NOFRAMES too.

Requirement A6: The system should not be browser / technology dependant.

7) Ensure user control of time-sensitive content changes

Anything that flashes, moves (quickly or not) is to be very bad practice. They are not considered at all accessible, and screen readers cannot interpret them.

The BLINK and MARQUEE elements are not defined in any W3C HTML specification and should not be used.

Requirement A7: The system should not contain anything that flashes or moves.

8) Ensure direct accessibility of embedded user interfaces

Ensure that the user interface follows principles of accessible design: device-independent access to functionality, keyboard operability, self-voicing, etc.

When an embedded object has its "own interface", the interface (like the interface to the browser itself) must be accessible. If the interface of the embedded object cannot be made accessible, an alternative accessible solution must be provided.

Requirement A8: Any embedded objects must also follow these accessibility guidelines.

9) Design for device-independence

Device-independent access means that the user may interact with the user agent or document with a preferred input (or output) device - mouse, keyboard, voice, head wand, or other.
Generally, pages that allow keyboard interaction are also accessible through speech input or a command line interface.

How to achieve this:

- For scripts, specify logical event handlers rather than device-dependent event handlers.
- Provide a logical tab order.
- Provide keyboard shortcuts to important links. E.g. HTML – accesskey.

Requirement A9: The system should allow users to interact with it in whichever way they choose.

10) Use interim solutions

Use interim accessibility solutions so that assistive technologies and older browsers will operate correctly.

How to achieve this:

- Do not cause pop-ups or other windows to appear and do not change the current window without informing the user.
- Labels must immediately precede their control on the same line.
- Use place-holding characters in edit boxes and text areas.
- Do not allow empty controls as old browsers do not cope well with this.

Requirement A10: The system must work with older browsers and assistive technologies.

11) Use W3C technologies and guidelines

These have built in accessibility features.

Also, make sure that other file formats (e.g. pdf's) transform into W3C techology gracefully – i.e. HTML. If not, provide an alternative format – e.g.rtf.

Requirement A11: If the system uses any non HTML file formats, ensure that they transform gracefully into W3C technology.

12) Provide context and orientation information

This is to help users understand complex pages or elements.
Grouping elements and providing contextual information about the relationships between elements can be useful for all users. Complex relationships between parts of a page may be difficult for people with cognitive disabilities and people with visual disabilities to interpret.

How to achieve this:

- Title each frame, its purpose and how they all relate.
- Divide large blocks of information into more manageable groups where natural and appropriate. (e.g. option groups).
- Associate labels explicitly with their controls

Requirement A12: The application must present information grouped appropriately for readability, with associated labels.

13) Provide clear navigation mechanisms

Provide clear and consistent navigation mechanisms to increase the likelihood that a person will find what they are looking for at a site.
E.g. orientation information, navigation bars, a site map, etc

How to achieve this:

- Make “Link text” meaningful.
- Provide metadata
- Provide information about the general layout of a site (e.g., a site map or table of contents).
- Provide navigation methods, and be consistent.
- If search functions are provided, enable different types of searches for different skill levels and preferences.
- Place distinguishing information at the beginning of headings, paragraphs, lists, etc (This is known as “front loading” and is paticularly useful for speech synthesiser users).
- Provide a means to skip over multi-line ASCII art.

Requirement A13: The system must allow users to navigate around the website effectively.

14) Ensure that documents are clear and simple

Ensure that documents are clear and simple so they may be more easily understood.
Consistent page layout, recognizable graphics, and easy to understand language benefit all users. In particular, they help people with cognitive disabilities or who have difficulty reading.

How to achieve this:

- Use the clearest and simplest language appropriate for a site's content.
- Supplement text with graphic or auditory presentations where they will facilitate comprehension of the page.
- Create a style of presentation that is consistent across pages.

Requirement A14: The system must use clear and simple language to aid understandability and readability.

Appendix J Summary of Design Patterns

The design patterns used come mainly from a website (www.welie.com) which is widely recommended from various other web sources (including university sites) and offers clear explanations and examples. The author has also written several papers in this field.

Please note that much of this material is lifted directly from the source. This appendix forms a point of reference for the system design structure.

1) Web based application

Problem: Users need to perform complex tasks on a web site

Use when: The site is for 'doing' things rather than simply providing information. In most cases there are 'objects' involved that belong to the users which the users need to create, change, delete or update. A web-based application is an application that could just as well be a normal application. It now just runs in a web browser.

Solution: Structure the site around 'views' and allow users to work inside views

The view provides a "safe" place where the users always return to after doing something using a form or wizard. The views are usually lists or tables that allow the display of information to be controlled.
Since web-based applications can be quite complex there is often help information.
A web-based application also usually has a simple navigation system that allows users to switch between views. A simple horizontal/vertical menu will usually suffice.

Why: Views contain the objects of interest and the view should therefore also be labelled according to the objects rather than the actions, although actions such as “Logout” are also often included. Structuring the web-based application mainly on views makes it easy for users to understand what they can do and how to interact with it.

2) Homepage

Problem: Users need to understand if they are at the right place, and if so, how they can move on to accomplish their task at your site

Use when: For most users, it will be the starting point for navigating through your site.

Solution: Create a homepage that introduces the site to users and that helps them to get on their way on the site

Users that found your site intentionally need feedback to confirm that they are in the right place. For other users that don't know you very well, you need to make clear who is behind the site and what there is to find. The homepage must balance navigation, branding, content, and promotion elements. Be careful not to make your page too full, users can only notice a couple of things at a time.

The homepage has three main functions:

- Introduction: The site must be introduced, both its purpose and identity. Users must know almost immediately what the site is for and who is behind it. For unknown brands or sites with a highly specialist purpose, a textual introduction is best. Another aspect of establishing an identity is having an “About Us” reachable from the navigation.

- Entrance: Users seldom find what they need on the homepage. Therefore, the site's main navigation must be clearly shown on the homepage so that users know where to get started. Other important elements include a search box, login if needed, and so on.
- Announcement: On most sites, things happen and the site changes. The homepage is to place to communicate what is new, what is going on, new promotions and so on.

The homepage is a special page. It is therefore quite normal that it has a slightly different layout than the other pages of the site. Nonetheless, some level of consistency should be retained so that the homepage and other pages clearly "belong" to the same site.

Why: The homepage is all about making a first impression.

3) Login

Problem: The users need to identify themselves so that stored data about/of them can be used in the process they are in.

Use when: When users frequently return to a site that uses large amounts of data about or belonging to the users, it is convenient to have users enter that information once and use it again for future visits to the site. In order to be able to access their data, users must complete “Registration” first.

Solution: When needed, ask the users to login using a combination of an email-address and a password

Late login: Allow users free access of the site until it is absolutely necessary that they identify themselves. Tell them why they need to log in.

Email address and password: Use a combination of an email address and a password. Optionally the email can be filled in automatically the next time the user returns. By using the email address as "login name" users can retrieve their password if they loose it (which they will...). Offer help to users who forgot their password.

Storing username/password locally: If the users need to log in very often, it can become annoying to type in the username/password over and over again. To help users, offer the possibility to remember the username/password locally on the users' computer.

Security: For some sites it is very important that users understand that their activities are safe enough. If so, tell users that they can use "secure" connections.

Provide feedback: Once users are logged in, let the site provide feedback that confirms this.

The login screen usually starts with a statement that the user needs to log in, then the username and password fields are shown. Users can use the TAB key to go from the username field to the password field and press ENTER instead of selecting the "Log in" button. A link to help users who have forgotten their password must be displayed directly under the password field.
Then an “Action Button” is shown to confirm the login.

Some users simply enter a username/password without having registered before, expecting the system to recognize this and respond intelligently. However, this usually results in an error message but you could also try to start the registration.

Why: Users do not like to be bothered with login procedures. This also facilitates browsing and exploring the site without commitments. Using a combination of the email address as the username and password makes it possible to email users their password when they loose it, and relieves them from remembering again another login name.

4) Action buttons

Problem: Users need to take important action.

Use when: Links to other pages or scripts are important actions because they can cause side-effects that cannot be undone easily or because they have other consequences for the users. Typically actions like "buy", "bid", "search", "add to cart" etc are important because they are part of the main task sequence that is relevant on the page.

Solution: Use push-button with the action 'verb' as part of the label.

A push-button visually stands out on the page and is easily distinguishable from a textual link. It therefore attracts more attention than simple text link and suggests an "action" because of it's "push-button" affordance. The button can be a standard system button but more frequently it is a graphic that fits in with the overall page design.
The larger they are, the easier it is to select them. The label inside the button must contain the 'action verb' as part of the label e.g. 'buy', 'add', 'submit' etc.

Place the push-button very near to the object(s) it belongs to. Quite often, the best places are above and to the right of the item itself. That way, you are can be sure that the button is visible without scrolling. Buttons used on a “Form” are an exception to this and the button is placed at the right-bottom location.

Why: The main reason for using action buttons is that because of their visual appearance, they get more attention and become visually distinct from normal text links.

5) Home link

Problem: Users need to get back to a safe/start/familiar point

Use when: In most Web/WAP sites there is a home page. This page is usually the starting point for interaction. When users arrive at an arbitrary page within the site, they should always be able to get back to the home page.

Solution: Use a fixed element, such as the site's logo, as a link to the home page

Add a link to the homepage on every page. Place the link at the top of the page and if appropriate on the bottom of the screen. If the site has a logo, make the logo a link to the homepage as well. Otherwise, simply link the homepage using the label "home", either a text label or an icon of a house. Make sure the link is always found on the same place.

Why: A home link provides a safe exit on every page and always on the same position. No matter what, the users can always get back to a familiar place. Logo's identify the site and are therefore easily chosen to get back to the home page.

6) Navigation (Also known as “bread crumbs”)

Problem: The users need to know where they are in a hierarchical structure

Use when: Sites with a large hierarchical information structure, typically more than 3 levels deep. The site has got a main navigation scheme that allows users to traverse the hierarchy. Users may want to jump several steps back instead of following the hierarchy. Users may be unfamiliar with the hierarchical structure of the information. Users may need to know where they can go.

Solution: Show the path from the top level to the current page.

E.g. Home > Next Level > Next Level > **Current Page**

The path shows the location of the current page in the total information structure. Each level of the hierarchy is labelled and functions as a link to that level. The current page is marked in order to give the users feedback about where they are now.

The path shows that a top-down path is traversed by using appropriate separators such as > or \ that suggest a downward motion. The path is placed in a separate "bar" that preferably spans the entire width of the content area, and is preferably placed above it.

Why: The bread crumbs show the users where they are and how the information is structured. Because users see the way the hierarchy is structured they can learn it more easily. By making each label a link, the users can quickly browse up the hierarchy. They take up minimal space on the page and leave most of the space for the real content.

7) Directory Navigation

Problem: Users need to select an item out of a set

Use when: There are several groups of items. The user may want to change between items in the same group. Users want to have an overview of what is all the sections.

Solution: Sum up level 1 and 2

Place all items of the second level under headings of the first level. Add the number of items present in each group. When the user has selected an item, the items of that group are shown with extra navigation at the top of the page.

Why: The directory provides an overview while at the same time showing the structure of the items. By always keeping the links to items within the same group visible, users can switch easily between those items.

8) Wizard

Problem: The user wants to achieve a single goal but several decisions need to be made before the goal can be achieved completely - which may not be known to the user.

Use when: A non-expert user needs to perform an infrequent complex task consisting of a small number of subtasks where decisions need to be made in each subtask. The user wants to reach the overall goal but may not be familiar or interested in the steps that need to be performed. The task can be ordered but are not always independent of each other.

Solution: Take the user through the entire task one step at the time. Let the user step through the tasks and show which steps exist and which have been completed.
When the complex task is started, the user is informed about the goal that will be achieved and the fact that several decisions are needed.
If the user cannot start the next task before completing the current one, feedback is provided indicating the user cannot proceed before completion.

The user should also be able to revise a decision by navigating back to a previous task.

Why: The navigation buttons suggest the users that they are navigating a path with steps. Each task is presented in a consistent fashion enforcing the idea that several steps are taken. When users are forced to follow the order of tasks, users are less likely to miss important things and will hence make fewer errors.

9) Processing Screen

Problem: Users need feedback that their action is being performed, but it may take a while to complete.

Use when: You are designing a site which has slow back-end systems connected to it. Some requests to the back-end system may take 5 to 30 seconds to complete and the users need some feedback telling them that their request is being performed and that they'll have to wait a bit. Only use this pattern when it is not possible to speed up the back-end processing time.

Solution: Provide a feedback page with animation

Provide information about the reason for the slow response so that users can have understanding of the problem. Also add an animation or real progress feedback so that users get a sense of continuity or progress.

Why: Although it would be best to provide real progress feedback, this is often not technically possible in a web environment. Providing this type of feedback is the least that should be done for users who need to wait.

10) Alternating Row Colours

Problem: Users need to read a lot of data in a table

Use when: The table can be complex and large, and may have several columns which make it hard to see which items belong to the same row.

Solution: Use alternating row colours for making the table more readable.

Use two colours of low saturation, e.g. white and a very light shade of an other colour that are only slightly different. Colour each row of the table by alternating these two colours. The effect can be enforced by adding another horizontal line using a dark colour.

Why: The row colours eliminate the need of table borders and make it easy for the eye to read a row. In a vertical sense, the colours make it easier to 'catch' an item because it is on either one of the colours.

11) Forms

Problem: Users need to provide personal information and send it to a service provider

Use when: Users need to provide information. Giving particular information must be part of a user task or at least provide benefit for the end users.

Solution: Offer users a form with the necessary elements

A form is essentially a collection of labels and input fields on a single page. When designing forms, there are several issues to be taken into account.

Make sure that users understand what you are asking from them. Give examples to re-enforce the meaning of the field. Use prompts sparely, adding more text also increases the chance that people won't read it. So keep any form of introduction text short, no more than a couple of lines.

Place elements in a logical ordering and group fields that together describe an entity, e.g. name and address could form "personal information".

Put the label left of the element or above it there are sever space limitations. Right align the label with the field so that label and field are always closely together.

In general making a distinction between mandatory and optional fields is a bad idea. Users should never have to fill in anything that is not required for the task at hand. However, there are certainly exceptions where optional fields make sense. In such cases it is important that it is clear to the user how filling in these fields will benefit them. If you have mandatory fields and optional fields, mark the mandatory fields with an asterisk "*".

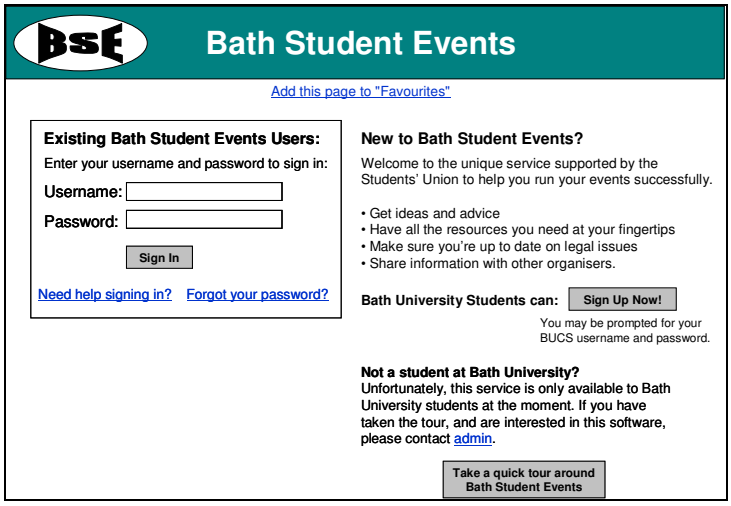
It is important to use the right input element for a certain field. This depends on the number of options, single/multiple choice, and there are radio buttons, list boxes and special controls available.

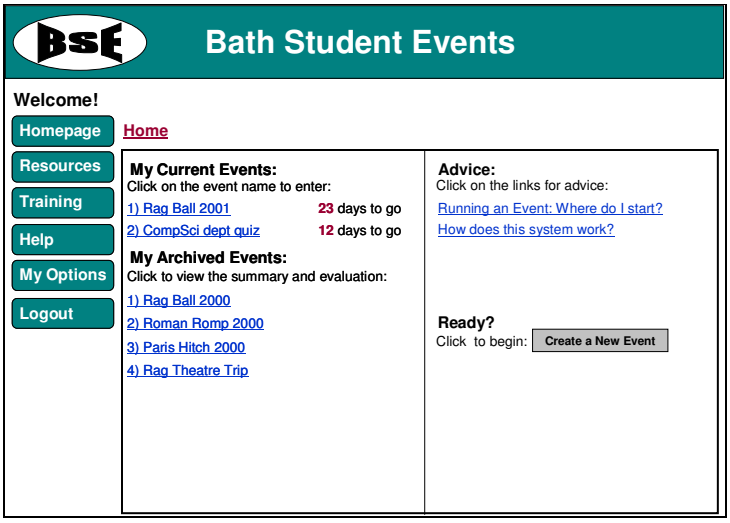
In order to speed up data entry, it is often good to use appropriate default values.
Preventing input errors Consider constraining the input to make sure users cannot provide invalid data.

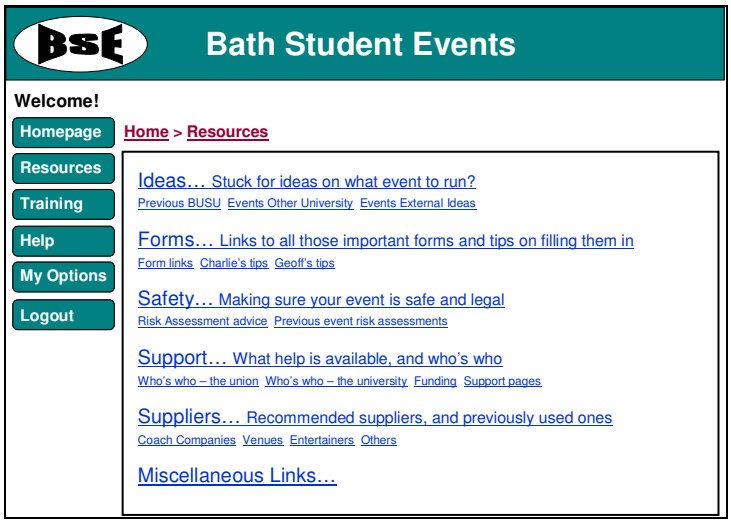
Filling in forms is tedious and it goes much faster if you can use the keyboard to go from one field to the other. Make sure that the “TAB” key can be used to do this and that the “ENTER” key is a shortcut for "confirm", submit, save etc.

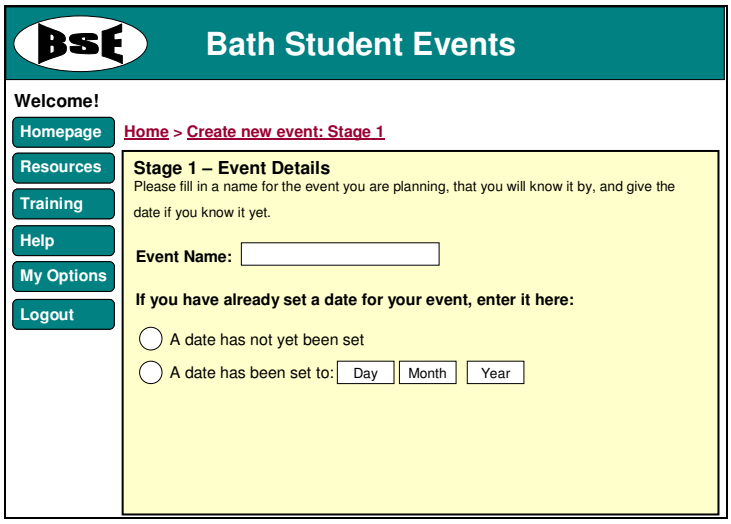
Why: Filling in forms is error-prone and things must be made as clear as possible for users. Using the right labels, widgets and defaults all contribute to the successful completion of the form.

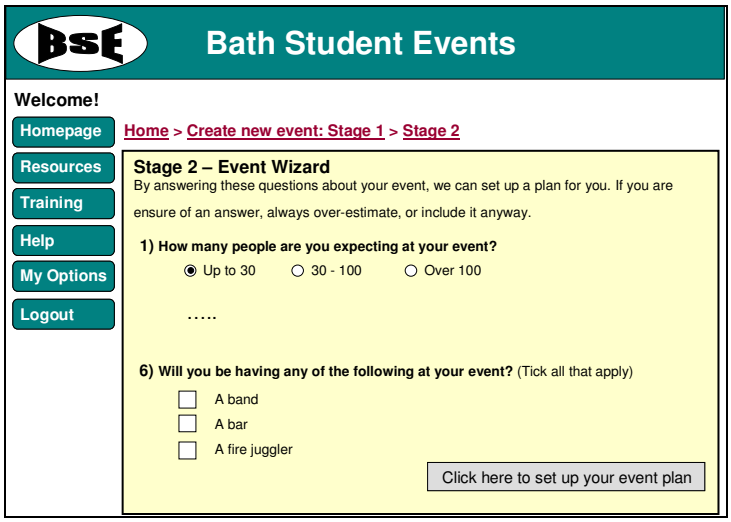
Appendix K User Testing Mock-ups

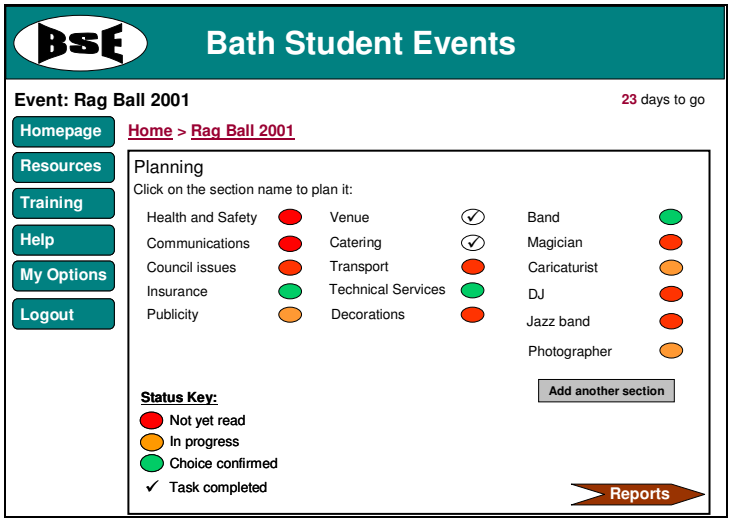


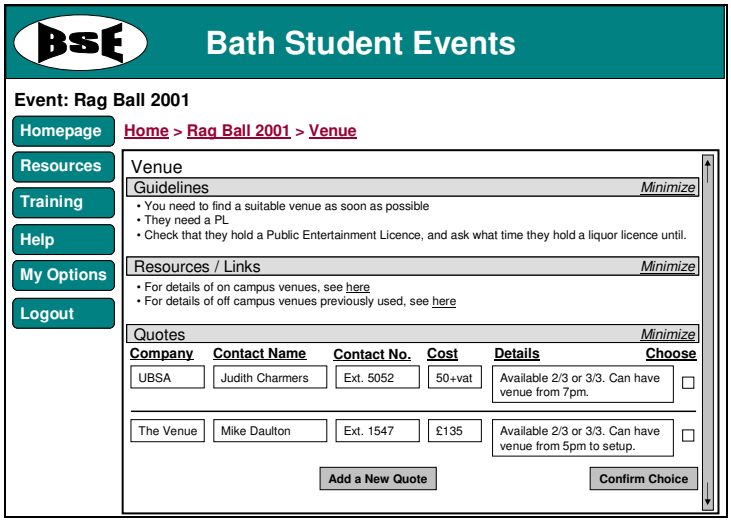


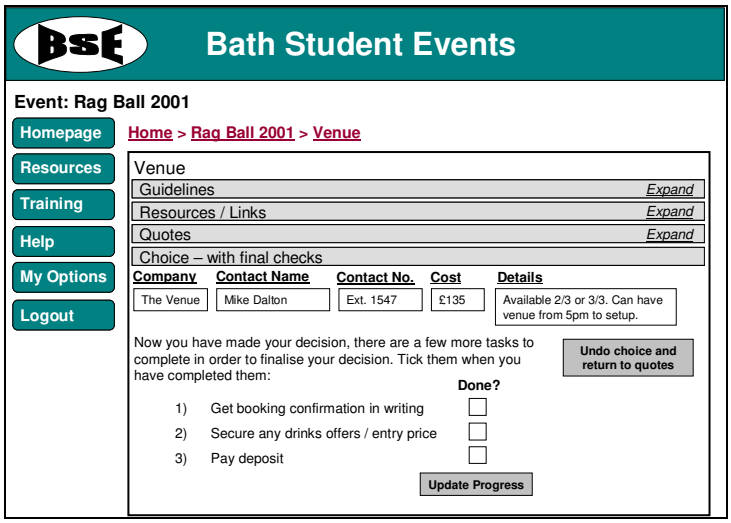



Bath Student Events
Welcome!
Homepage
Resources
Training
Help
My Options
Logout
Home > Create new event: Stage 1
Stage 1 – Event Details
Please fill in a name for the event you are planning, that you will know it by, and give the date if you know it yet.
Event Name:
If you have already set a date for your event, enter it here:
A date has not yet been set
A date has been set to: Day Month Year


Bath Student Events
Welcome!
Homepage
Resources
Training
Help
My Options
Logout
Home > Create new event: Stage 1 > Stage 2
Stage 2 – Event Wizard
By answering these questions about your event, we can set up a plan for you. If you are ensure of an answer, always over-estimate, or include it anyway.
1) How many people are you expecting at your event?
Up to 30
30 - 100
Over 100
.....
6) Will you be having any of the following at your event? (Tick all that apply)
A band
A bar
A fire juggler
Click here to set up your event plan


Bath Student Events
Event: Rag Ball 2001
23 days to go
Homepage
Resources
Training
Help
My Options
Logout
Home > Rag Ball 2001
Planning
Click on the section name to plan it:
Health and Safety
Communications
Council issues
Insurance
Publicity
Venue
Catering
Transport
Technical Services
Decorations
Band
Magician
Caricaturist
DJ
Jazz band
Photographer
Add another section
Status Key:
Not yet read
In progress
Choice confirmed
Task completed
Reports


Bath Student Events
Event: Rag Ball 2001
Homepage
Resources
Training
Help
My Options
Logout
Home > Rag Ball 2001 > Venue
Venue
Guidelines
Minimize
• You need to find a suitable venue as soon as possible
• They need a PL
• Check that they hold a Public Entertainment Licence, and ask what time they hold a liquor licence until.
Resources / Links
Minimize
• For details of on campus venues, see here
• For details of off campus venues previously used, see here
Quotes
Minimize
Company
Contact Name
Contact No.
Cost
Details
Choose
UBSA
Judith Charmers
Ext. 5052
50+vat
Available 2/3 or 3/3. Can have venue from 7pm.
The Venue
Mike Daulton
Ext. 1547
£135
Available 2/3 or 3/3. Can have venue from 5pm to setup.
Add a New Quote
Confirm Choice


Bath Student Events
Event: Rag Ball 2001
Homepage
Resources
Training
Help
My Options
Logout
Home > Rag Ball 2001 > Venue
Venue
Guidelines
Expand
Resources / Links
Expand
Quotes
Expand
Choice – with final checks
Company
Contact Name
Contact No.
Cost
Details
The Venue
Mike Dalton
Ext. 1547
£135
Available 2/3 or 3/3. Can have venue from 5pm to setup.
Now you have made your decision, there are a few more tasks to complete in order to finalise your decision. Tick them when you have completed them:
Undo choice and return to quotes
Done?
1) Get booking confirmation in writing
2) Secure any drinks offers / entry price
3) Pay deposit
Update Progress

Appendix L Testing Theory

This appendix outlines the theory behind the testing strategy. It describes the types of tests that were carried out and the rationale behind them.

Unit testing

Breaking down the application into smaller segments to test individually, makes overall testing a lot easier and simpler, and is what unit testing aims to do.

Bruegge (2004) identifies several sections of unit testing that provide a complete framework for assessing system components:

1) **Equivalence testing:** is a form of black box testing which assumes that values in the same "set" will be treated by the system in the same way. This also helps to reduce the amount of time required for testing. These sets include invalid data to try and cause an error.
2) **Boundary testing:** tests values on the boundaries on the equivalence classes.
3) **Path testing:** is a form of white box testing that checks that every piece of code is executed during the tests.

By breaking down the tests further and grouping test values into relevant test cases, it makes it a lot easier to identify faults and hence correct them. Test stubs are often used where possible to make it even easier to identify the source of a problem. Test stubs simulate components or values that are called by the component being tested, but this is not always an easy task. This testing strategy will limit the use of test stubs to simply hard-coded values to assist in the problem identification process.

The easiest example to explain the unit testing framework is the tests for a screen requiring data entry, as seen in tests 5.4 – 5.23.

This framework of test cases will be used throughout the unit testing

This application consists of a mySQL database, and the php front end. The bottom-up approach dictates that each for a particular unit, each will be tested in turn before they are tested together.

Stage 1: Database testing

The SQL statements which are used by the application will first be checked using an mySQL front end graphical user interface called SQLyog. Fields that will contain application driven values will be replaced with absolute values for the purpose of this testing, which aims to check the validity of the SQL.

Example: SELECT vEventName FROM EVENT WHERE ipkEventID = 5

Stage 2: PHP Front end testing

All parts of the application will be checked to ensure they are working correctly and as expected. Dynamic SQL statements that are created will be displayed on screen to ensure that these are correct also.

SQL Example: SELECT g.vGuidelineText as GuidelineText FROM EVENTtoGUIDELINESET egs, GUIDELINES g WHERE egs.iGuidelineID = g.ipkGuidelineID AND egs.iGuidelineSetID="**.** $GuidelineSetID
A simple php echo statement will be used to check that the $GuidelineSetID is being used correctly.

Stage 3: Database and PHP relationship testing

Once stages 1 and 2 are complete, a final test will be run to ensure that the database back-end is working correctly with the front end application. These consist of those areas in which the front end requests the execution of SQL code (which it may or may not have generated itself) which the database then runs, and returns the results back to the front end.

Integration Testing

Unit testing seeks to check each system component individually, leading on to this next stage of integration testing to see how components work together.

My approach will test components together in small groups at first, which will then grow as tests are completed.
If there are 3 components (A, B, and C) A and B will be tested together, B and C will be tested together and then A, B and C will all be tested together.
Strictly speaking, for absolute completeness, A and C would also be tested together before the final test, but due to the constraints on time within this project, reliance will be put on the final test to detect an anomalies.

Dummy values (basic test stubs) will be used to make testing more simple.

System testing

Bruegge (2004) defines the high level areas of system testing as:

1) **Requirements testing:** comparing the system to its functional requirements.
2) **Performance testing:** comparing the system to its non-functional requirements.
3) **Pilot testing:** invites potential users to give feedback of the overall performance of the system.
4) **Acceptance testing:** are performed by the customer usually in the development environment to see if it meets their criteria – as usually set out in the formal project agreement.
5) **Installation testing:** are performed by the customer usually in their own environment to see if it meets their criteria.

Once again, as this is a prototype that is being tested, only stages 1 and 2 apply here. User related testing will be discussed in section 7.6.

Requirements testing

Requirements' testing quite simply ensures that the complete system complies with the functional requirements of the system laid out.
Both the requirements directly listed,
Test cases are derived from the use case models specified in section 4.8 and are used in addition to those requirements directly listed to judge if the system meets the users overall needs.

Performance testing

Performance testing compares the application to the stated non functional requirements. Although many parts of this do not apply to the prototype developed, they are included here for completeness.
Again, these look at Bruegge's (2004) outline and how each applies to this prototype testing.

Stress testing

These tests aim to cause the system to error by sending lots of requests to it, and forcing it to do a lot of things at the same time. The prototype has not been designed for such activity so it would not be a surprise to see a fault occur.

Timing tests

These measure the systems performance by timing how long it takes to complete each task.
Due to the internal nature of this project, the database is being hosted on a Bath University server, and the pages on a Bath internal private directory. Hence for this prototype, I have no control over the resources available. The times taken to complete tasks will also depend greatly on the demand on the server and how many processes are being run concurrently.
If this project was to be developed further, this type of testing would then become essential.

Security testing

Security tests are usually completed by people familiar with typical system security flaws, to try to break into the system. The only confidential information this system stores, is the user's personal information and it is important that it is kept that way. A final implementation of this project would have to take data protection very seriously, and any personal information declared to the Data Protection Commissioner.

Recovery tests

The recovery tests determine how this system would respond in the event of a serious error. Examples of these are hardware failure, and if the users' browser crashes in the middle of a task. These can be seen in tests 10.1 – 10.2.

Appendix M Test Plan

Login page

Test #	Test	Expected result	Actual result (if different)	Test passed?	Comments
1.1	The title bar and toolbar are displayed correctly	Title bar and tool bar appear correctly		Yes	
1.2	The message bar contains the correct value	No text is displayed – simply a gap is displayed below the title bar		Yes	
1.3	Click the login button without entering a username	Pop up window asking the user to enter a username and password	Pop up window only appeared when using IE and Netscape	No	This is not a critical failure, nothing happens using the other browsers. It should be self explanatory
1.4	Click the login button without entering a password	Pop up window asking the user to enter a username and password	Pop up window only appeared when using IE and Netscape	No	This is not a critical failure, nothing happens using the other browsers. It should be self explanatory
1.5	Enter an incorrect username and password combination	Page reloads with error message and hints for the user		Yes	
1.6	Enter a correct username and password combination	User is re-directed to their personal homepage		Yes	
1.7	Enter a correct username and password but with incorrect cases (caps)	Page reloads with error message and hints for the user	User is logged in	No	This should be corrected if this project is taken further. It is not a critical failure for this prototype

Personal Homepage

2.1	The title bar and toolbar are displayed correctly	Title bar and tool bar appear correctly		Yes	
2.2	The message bar contains the correct value	The text “Welcome to Bath Student Events” is displayed		Yes	

2.3	The breadcrumbs are setup correctly – clicking on them yields the expected results	"Home" is displayed which links to this page when it is clicked – in essence it refreshes it		Yes	
2.4	All events related to the current user only are displayed	All event names displayed		Yes	
2.5	These event names are links to the planning page of that event	Clicking on the event name will take the user to the planning page of that event		Yes	
2.6	The number of "days to go" / "date passed" / "date not set" is correct	Correct values are displayed		Yes	
2.7	If no events are associated, descriptive text appears	The message is displayed		Yes	
2.8	The link appears to setup a new event, which is active	On clicking on this link, users are re-directed to the event setup wizard		Yes	

Event Plan Index

3.1	The title bar and toolbar are displayed correctly	Title bar and tool bar appear correctly		Yes	
3.2	The message bar contains the correct value	The text "Event: [Event name]" is displayed		Yes	
3.3	The breadcrumbs are setup correctly – clicking on them yields the expected results	"Home" is displayed which links to the homepage, the link to [Event Name] reloads the current page		Yes	
3.4	Each relevant event planning section is listed	Links are displayed correctly with the correct section name		Yes	
3.5	These sections are links into each section	Clicking on each link re-directs the user to the relevant section		Yes	

Plan Event Section

4.1	The title bar and toolbar are displayed correctly	Title bar and tool bar appear correctly		Yes	
4.2	The message bar contains the correct value	The text “Event: [Event name]” is displayed		Yes	
4.3	The breadcrumbs are setup correctly – clicking on them yields the expected results	“Home” is displayed which links to the homepage, the link to [Event Name] reloads the planning page, and [section name] which reloads the current page		Yes	
4.4	The correct guidelines are displayed	The correct guidelines are displayed		Yes	
4.5	The correct resources are displayed	The correct resources are displayed		Yes	
4.6	If there are no resources, then text is displayed	“There are currently no resources available for this section” is displayed below the title for Resources		Yes	
4.7	Click on the link to add notes (only available if no notes are currently available)	A box to enter notes in is displayed, along with buttons to update and delete them		Yes	
4.8	Currently stored notes are displayed	If notes have already been entered, they are displayed in the box along with buttons to update and delete them		Yes	
4.9	Click on button to delete notes	Notes are deleted and the user returned to the plan section page, with the link to add notes restored		Yes	
4.10	Click on button to update notes	Notes are updated		Yes	It is not obvious that this action has taken place.
4.11	Existing quotes displayed correctly	Existing quotes displayed correctly		Yes	
4.12	Click on button to delete a quote	Quote is deleted and user returned to the planning page		Yes	

4.13	Correct buttons are displayed	If no quotes have been entered, only the “add new quote” button is displayed		Yes	
4.14	Click on the add new quote button	Form for the user to fill in quote data is displayed		Yes	
4.15	After the previous action, click the add new quote button again (without having saved the previous one)	No difference. – the same form is displayed, and only once		Yes	
4.16	Click on save new quote	New quote saved and user is returned to the planning section page with the new quote included in the display		Yes	
4.17	Click on update all quotes button	All quotes are updated and the user is returned to the planning section page		Yes	It is not obvious that this action has taken place.
4.18	Click on confirm choice of quotes button with no quotes selected	Nothing		Yes	An improvement would be if the user was given an error message
4.19	Click on confirm choice of quotes button with just one quote selected	Page changes to the post choice page with the chosen quote displayed		Yes	
4.20	Correct to do list options are displayed	Correct to do list options are displayed		_	Note: This is not an active feature, merely for display only. Further development would require this feature to be properly implemented.
4.21	Headings of sections are displayed	Headings of sections are displayed		Yes	
4.22	Click on section header links to expanding the section	Section information is revealed		No	This was not completed. These links merely display the potential capability
4.23	Click on undo choice	Choice is undone an the user is returned to the plan section page looking directly at the		Yes	

		quotes section			
4.24	Repeat previous 5 tests whilst selecting more than one quote to confirm	Results the same as the previous tests		Yes	

Event Wizard – Stage One

5.1	The title bar and toolbar are displayed correctly	Title bar and tool bar appear correctly		Yes	
5.2	The message bar contains the correct value	The text “Set up a new event” is displayed		Yes	
5.3	The breadcrumbs are setup correctly – clicking on them yields the expected results	“Home” is displayed which links to the homepage, “create new event stage 1” reloads the current page		Yes	
5.4	Click submit with no event name entered	The user is returned to this page with a related error message about the event name		Yes	
5.5	Enter a event name of 1 character: “f”	The value is accepted and the user progresses to stage 2 of the wizard		Yes	
5.6	Enter a event name of 1 character: “2”	The value is accepted and the user progresses to stage 2 of the wizard		Yes	
5.7	Enter a event name of 6 characters: “social”	The value is accepted and the user progresses to stage 2 of the wizard		Yes	
5.8	Enter a event name of 13 characters: “freshers week”	The value is accepted and the user progresses to stage 2 of the wizard	The value was not accepted, the user was given an error message.	No	This is a critical error – spaces in event names are very common and the system must allow for this. Adjustments were made.
5.9	Repeat the 5 previous tests	The system performs as expected		Yes	Re-running these tests ensure that nothing else has changed

					as a result of the correction in the code
5.10	Enter a event name of 29 characters: “aaa 222 vvvvvvv bbbbbbb ddddd” (boundary test)	The value is accepted and the user progresses to stage 2 of the wizard		Yes	
5.11	Enter a event name of 30 characters: “ssssss 791 ttttt uuu ooo eeeee” (boundary test)	The value is accepted and the user progresses to stage 2 of the wizard		Yes	
5.12	Enter a event name of 31 characters: “ttt 77810 jjjjj fff ssss eee uu” (boundary test)	The value was not accepted, the user was given an error message.		Yes	
5.13	Enter a event name of 1 character: “ ” [space] – checks code alterations	The value was not accepted, the user was given an error message.		Yes	
5.14	Enter a event name of 6 characters beginning with a space: “ tttt d”– checks code alterations	The value was not accepted, the user was given an error message.		Yes	
5.15	Enter a date which is valid: “1st January 2004” (along with a valid username)	The date is accepted and the user progresses to stage 2 of the wizard		Yes	
5.16	Enter a date which is valid: “15th September 2004” (along with a valid username)	The date is accepted and the user progresses to stage 2 of the wizard		Yes	
5.17	Enter a date which is valid: “30th September 2004” (boundary test) (along with a valid username)	The date is accepted and the user progresses to stage 2 of the wizard		Yes	
5.18	Enter a date which is invalid: “31st September 2004” (along with a valid username)	The value was not accepted, the user was given an error message.		Yes	
5.19	Enter a date which is invalid: “31st June 2004” (along with a valid username)	The value is not accepted, the user is given an error message.		Yes	

5.20	Enter an invalid date whilst "no" is selected for the event having a date	The information is accepted and the user progresses to stage 2 of the wizard	The value was not accepted, the user was given an error message.	No	This is difficult, students have to actively choose date to reach an invalid one (as the default is 1st January 2004) This setup could cause confusion and requires re-design for future work
5.21	Enter a date which is valid: "29th February 2004" (along with a valid username)	This tests the leap year settings. The information is accepted and the user progresses to stage 2 of the wizard		Yes	
5.22	Enter a date which is invalid: "29th February 2005" (along with a valid username)	This tests the leap year settings. The value is not accepted, the user is given an error message.		Yes	
5.23	Enter an invalid date: "30th February 2004" (along with a valid username)	Check other February dates - The value is not accepted, the user is given an error message.		Yes	

Event Wizard – Stage Two

6.1	The title bar and toolbar are displayed correctly	Title bar and tool bar appear correctly		Yes	
6.2	The message bar contains the correct value	The text "Set up a new event" is displayed		Yes	
6.3	The breadcrumbs are setup correctly – clicking on them yields the expected results	"Home" is displayed which links to the homepage, "create new event stage 1" links to the previous stage, "stage 2" refreshes the current page		No	Clicking on "stage 2" causes problems, as the previous stage is a form which posts to this page. This action is disabled for now.
6.4	Re run previous test following coding change	"Home" is displayed which links to the homepage, "create new event stage 1" links to the previous stage, "stage 2" is inactive text.		Yes	

6.5	Click on submit	The new event is created, and the user is re-directed to the planning index for the new event		Yes	
6.6	Submitting the options creates a new event with the right information	The new event is created correctly		Yes	

Help page

7.1	Click on “help” from the toolbar	The help index is displayed		Yes	
7.2	The title bar and toolbar are displayed correctly	Title bar and tool bar appear correctly		Yes	
7.3	The message bar contains the correct value	The text “Set up a new event” is displayed		Yes	
7.4	The breadcrumbs are setup correctly – clicking on them yields the expected results	“Home” is displayed which links to the homepage, “help” reloads the current page		Yes	
7.5	Click on the link “Email a question to Charlie”	The user is re-directed to the email page		Yes	

Email a question page

8.1	Click on the “help” button on the toolbar	The help index is displayed		Yes	
8.2	The title bar and toolbar are displayed correctly	Title bar and tool bar appear correctly		Yes	
8.3	The message bar contains the correct value	The text “Set up a new event” is displayed		Yes	
8.4	The breadcrumbs are setup correctly – clicking on them yields the expected results	“Home” is displayed which links to the homepage, “help” links back to the help index, “Email a question to Charlie” reloads the current page		Yes	

8.5	The “event name” text box displays a relevant event	This box displays the event that was last looked at. If no event has been looked at yet, it is empty		Yes	
8.6	The “event date” text box displays a relevant event	This box displays the date of event that was last looked at. If no event has been looked at yet, it is empty		No	If an event has not yet been looked at, today’s date is displayed.

Logout Function

9.1	Click on the “logout” button on the toolbar	The user is logged out and returned to the login plage		Yes	

Sample Recovery Tests

10.1	Whilst the application is setting up a new event, close the browser	The event set up successfully		_	This test proved to be impossible to complete as the application setup the event instantly.
10.2	Whilst the application is setting up a new event, turn off the computer	The event set up successfully		_	This test proved to be impossible to complete as the application setup the event instantly.

Appendix N Completed User Testing Questionnaires

Appendix O Code

In this appendix is a copy of all the code used in the implementation of the system grouped into two sections, those of functions and web pages.

Functions

Checklogin.php

```
<?php

   function redirect_to_login($redirect) {

     if ($redirect == 1)
     {
       header("Location: ../pages/login.php");
     }
   }

   function connect_to_db() {

     $db=mysql_connect("midge.bath.ac.uk","ma0car","FkMR722d");
     mysql_select_db("csprojStudentEvents");
   }

   //start of main code

   global $session_db;
   global $session_UserId;
   global $PHPSESSID;
   global $session_EventId;
   global $session_EventName;

   session_start();

   if ($_SESSION["session_UserId"]) {

     //header("Cache-control: private");  //rec by phpfreaks -
http://www.phpfreaks.com/tutorials/41/1.php

     //Sometimes the var is posted in the form, sometimes with the url
     $PHPSESSID = $_GET["PHPSESSID"];
     if (!$PHPSESSID) {
       $PHPSESSID = $_POST["PHPSESSID"];
     }

   } else {
     //echo "GO AWAY!";
     $redirect = 1;
   }

?>
```

connect.php

```
<?php
   $db=mysql_connect("midge.bath.ac.uk","ma0car","FkMR722d");
   mysql_select_db("csprojStudentEvents");
?>
```

date_eval.php

```
<?php
   /*This function is lifted from: http://uk.php.net/date - 7th March 2004
   - It calcs the diff in days between 2 dates - does also go negative*/
      function date_diff($dat1,$dat2)
      /* Dat1 and Dat2 passed as "YYYY-MM-DD" */
      {
        $tmp_dat1 = mktime(0,0,0,
          substr($dat1,5,2),substr($dat1,8,2),substr($dat1,0,4));
        $tmp_dat2 = mktime(0,0,0,
          substr($dat2,5,2),substr($dat2,8,2),substr($dat2,0,4));

        $yeardiff = date('Y',$tmp_dat1)-date('Y',$tmp_dat2);
        /* a leap year in every 4 years and the days-difference */
        $diff = date('z',$tmp_dat1)-date('z',$tmp_dat2) +
          floor($yeardiff /4)*1461;

        /* remainder */
        for ($yeardiff = $yeardiff % 4; $yeardiff>0; $yeardiff--)
        {
          $diff += 365 + date('L',mktime(0,0,0,1,1,intval(substr((($tmp_dat1>$tmp_dat2) ? $dat1 :
$dat2),0,4))-$yeardiff+1));
        }
        return $diff;
      }

     //I wrote this one to give me something meaningful from the date diff result, and take the
complexity out of the main body for ease of reading.
     function Eval_DateDiff($func_DaysToGo)
     {
       If ($func_DaysToGo < 0)
       {
         return "DATE PASSED";
       } elseif ($func_DaysToGo == 0) {
         return "TODAY!";
       } else {
         return $func_DaysToGo." days to go";
       }
     }
?>
```

header.php

```
<table border="1" bordercolor = "black" cellpadding="0" cellspacing="0" width="656"  align="center"
bgcolor="#006666">
   <tr>
     <td width ="20%" height ="47" align = "center">
       <?php
         echo "<a href=\"../pages/homepage.php?PHPSESSID=".$PHPSESSID."\">";
       ?>
       <img border="0" src="../images/logo3.jpg" width="107" height="47" alt="Bath Student Events
Logo">
       </a>
     </td>

     <td width ="60%" align = "center">
       <font class="bigwhitetitle">Bath Student Events
     </td>

     <td width ="20%" align = "center">
       <font class="white">
       <?php
       $todaysdate=date("D dS F");
       echo $todaysdate;
       ?>
     </td>
   </tr>
</table>
logout.php
<?php

//de-register and destroy session
//from php.net example code

session_start();
```

```
// Unset all of the session variables.
session_unset();
// Finally, destroy the session.
session_destroy();

//return user to the login page

header("Location: ../pages/login.php");
?>
```

messagebar.php

```
<table border="0" bordercolor = "black" cellpadding="5" cellspacing="0" width="656"  align="center">
  <tr>
    <td width ="100%" align = "left">
      <font class="msgtitle"><strong>
        <?php
          //echo "The message should appear below: <br>";
          if ($str_message) {
            echo $str_message;
          } else {
            echo " ";
          }
        ?>
      </strong></font>
    </td>
  </tr>
</table>
```

postchoiceSetup.php

```
<?php

//This code is for demonstrating the system features only.
//Print guidelines heading
echo "<tr>";
  echo "<td bgcolor=\"silver\" valign=\"top\" width=\"500\">";
  echo "<font class = \"black\"><strong>";
  echo "Guidelines - ";
  echo "  ";
  echo "</font><font class = \"black\"><i><u>";
  echo "Click to Expand";
  echo "</u></i></td>";
echo "</tr>";

//Print resources heading
echo "<tr>";
  echo "<td bgcolor=\"silver\" valign=\"top\" width=\"500\">";
  echo "<font class = \"black\"><strong>";
  echo "Resources - ";
  echo "  ";
  echo "</font><font class = \"black\"><i><u>";
  echo "Click to Expand";
  echo "</u></i></td>";
echo "</tr>";

//Print notes heading
echo "<tr>";
  echo "<td bgcolor=\"silver\" valign=\"top\" width=\"500\">";
  echo "<font class = \"black\"><strong>";
  echo "Personal Notes - ";
  echo "  ";
  echo "</font><font class = \"black\"><i><u>";
  echo "Click to Expand";
  echo "</u></i></td>";
echo "</tr>";

//Print quotes heading
echo "<tr>";
  echo "<td bgcolor=\"silver\" valign=\"top\" width=\"500\">";
  echo "<font class = \"black\"><strong>";
  echo "Quotes - ";
  echo "  ";
  echo "</font><font class = \"black\"><i><u>";
  echo "Click to Expand";
  echo "</u></i></td>";
echo "</tr>";

//Print Choice heading
echo "<tr>";
  echo "<td bgcolor=\"silver\" valign=\"top\" width=\"500\">";
  echo "<font class = \"black\"><strong>";
  echo "Choice - with final checks";
  echo "</strong></font></td>";
echo "</tr>";

//Get chosen quotes
$Get_chosen_quotes_SQL = "SELECT * FROM EVENTtoCHOSENQUOTESET eqs, QUOTES q WHERE eqs.iQuoteID =
q.ipkQuoteID AND eqs.iChosenQuoteSetID=".$ChosenQuoteSetID;

$Get_chosen_quotes = mysql_query($Get_chosen_quotes_SQL);
$num_chosen_quotes= mysql_num_rows($Get_chosen_quotes);

for ($cq=0; $cq < $num_chosen_quotes; $cq++) {

//Scoop up the quote details ready to use below
$this_chosen_quote=mysql_fetch_array($Get_chosen_quotes);

$thisQuoteID = $this_chosen_quote["ipkQuoteID"];
$thisCompany = $this_chosen_quote["vCompany"];
$thisContactName = $this_chosen_quote["vContactName"];
$thisContactNumber = $this_chosen_quote["vContactNumber"];
$thisCost = $this_chosen_quote["vCost"];
$thisDetail = $this_chosen_quote["vDetails"];

echo "<tr><td>";

echo "<form name=\"input\" action=\"../functions/undoChoice.php\" method=\"post\">";
echo "<table>";
echo "<tr>";
  echo "<td width=\"60\" valign=\"center\"><font class = \"smalltitle\"><strong>";
  echo "Company:";
  echo "</td>";
  echo "<td width=\"60\" valign=\"center\"><input type=\"text\"  name=\"company_".$q."\"
value=\"".$thisCompany."\" align=\"center\" maxlength=\"25\" size=\"15\">";
  echo "</td>";
  echo "<td width=\"10\" valign=\"center\"><font class = \"smalltitle\"><strong>";
  echo "Cost:";
  echo "<td valign=\"center\"><input type=\"text\" value=\"".$thisCost."\" name=\"cost_".$q."\"
align=\"center\" maxlength=\"10\" size=\"5\">";
  echo "</td>";
  echo "<td align = \"center\">";
    echo "<input type=\"submit\" value=\"Undo choice\" name = \"undoChoice\" align = \"center\"/>";
  echo "</td>";
echo "</tr>";
echo "<tr>";
  echo "<td width=\"100\" valign=\"center\"><font class = \"smalltitle\"><strong>";
  echo "Contact:";
  echo "</td>";
  echo "<td width=\"60\" valign=\"center\"><input type=\"text\" name=\"contactName_".$q."\"
value=\"".$thisContactName."\" align=\"center\" maxlength=\"25\" size=\"15\">";
  echo "</td>";
  echo "<td colspan = \"2\" rowspan=\"2\"><TEXTAREA   NAME=\"quoteNotes_".$q."\" COLS=15 ROWS=3>";
  echo $thisDetail;
  echo "</TEXTAREA>";
  echo "</td>";
echo "</tr>";
echo "<tr>";
  echo "<td width=\"100\" valign=\"center\"><font class = \"smalltitle\"><strong>";
```

```
   echo "Contact no.:";
   echo "</td>";
   echo "<td width=\"60\" valign=\"center\"><input type=\"text\" value=\"".$thisContactNumber."\"
name=\"contactNumber_".$q."\" align=\"center\" maxlength=\"25\" size=\"15\">";
   echo "</td>";
echo "</tr>";

//Pass key variables as hidden variables
echo "<input type=\"hidden\" name=\"PHPSESSID\" value=\"".$PHPSESSID."\">";
echo "<input type=\"hidden\" name=\"EPSId\" value=\"".$EPSId."\">";
echo "<input type=\"hidden\" name=\"ChosenQuoteSetID\" value=\"".$ChosenQuoteSetID."\">";

echo "</table>";
echo "</form>";
echo "<hr>";
echo "</tr></td>";
}

//List to dos
echo "<tr>";
echo "<td colspan = \"3\"><font class = \"black\">";
echo "Now you have made your decisions, there are a few loose ends to tie up. Check each box, and
update your progress when necessary.";
echo "</td></tr>";

echo "<tr>";
echo "<td colspan = \"3\">";

echo "<form name=\"input\" action=\"UPDATE_PROGRESS\" method=\"post\">";
echo "<table>";
echo "<tr>";
echo "<td><font class = \"black\">";
echo "1) Get booking confirmation in writing";
echo "</td>";
echo "<td valign=\"top\" align=\"center\"><font class = \"smalltitle\"><strong>";
echo "<input type=\"checkbox\" name=\"todoDone1\" value=\"1\"/>";
echo "</td>";
echo "</tr>";
echo "<tr>";
echo "<td><font class = \"black\">";
echo "2) Secure any drinks offers / entry price";
echo "</td>";
echo "<td valign=\"top\" align=\"center\"><font class = \"smalltitle\"><strong>";
echo "<input type=\"checkbox\" name=\"todoDone2\" value=\"1\"/>";
echo "</td>";
echo "</tr>";
echo "<tr>";
echo "<td><font class = \"black\">";
echo "3) Pay deposit";
echo "</td>";
echo "<td valign=\"top\" align=\"center\"><font class = \"smalltitle\"><strong>";
echo "<input type=\"checkbox\" name=\"todoDone3\" value=\"1\"/>";
echo "</td>";
echo "</tr>";
echo "<tr>";
   echo "<td> </td>";
   echo "<td align = \"center\">";
     echo "<input type=\"submit\" value=\"Update Progress\" align = \"center\"/>";
   echo "</td>";
echo "</tr>";
echo "</table>";
echo "</form>";
echo "</td></tr>";
?>
```

setupplan.php

```
<?php
  include ("../functions/checklogin.php");
?>

<html>
<head>
  <title>Setting up the event plan</title>

  <?php
    redirect_to_login($redirect);
  ?>

  <LINK REL=StyleSheet HREF="../functions/styles.css" TYPE="text/css">

</head>
<body>

<h1>Check that the vars have been picked up</h1>
<?php

  //Make sure user is logged in, else deal with it
  //Open the database and the session
  connect_to_db();

  //Pick up the values that were submitted
  $numbers = $_POST["numbers"]; //Q1

  $susoc = $_POST["susoc"]; //Q2
  $oncampus = $_POST["oncampus"]; //Q3

  //Q4
  $ownfood = $_POST["ownfood"];
  $intcat = $_POST["intcat"];
  $extcat = $_POST["extcat"];
  //what are you going to do if this is checked and so are some of the catering ones?
  //Going to ignore this value if others are checked - after all, this value makes no difference to
anything.
  $nofood = $_POST["nofood"];

  //Q5
  $pubtour = $_POST["pubtour"];
  $theatre = $_POST["theatre"];
  $fayre = $_POST["fayre"];
  $political = $_POST["political"];

  //Q6
  $music = $_POST["music"];
  $tempstruct = $_POST["tempstruct"];
  $public = $_POST["public"];
  $road = $_POST["road"];
  $intbar = $_POST["intbar"];

  //Values from Stage 1
  $final_event_date = $_POST["final_event_date"];
  $eventname = $_POST["eventname"];

  //Put together the criteria string - open bracket first and put in this var - this will always
have an //influencial varialble
  $sCriteriaString = "(".$numbers;

  if ($susoc == 1) {
    $sCriteriaString = $sCriteriaString . ",4";
  }
  if ($oncampus == 1) {
    $sCriteriaString = $sCriteriaString . ",5";
  }
  if ($ownfood == 1) {
    $sCriteriaString = $sCriteriaString . ",6";
  }
  if ($intcat == 1) {
    $sCriteriaString = $sCriteriaString . ",7";
  }
  if ($extcat == 1) {
    $sCriteriaString = $sCriteriaString . ",8";
  }
```

```
  if ($pubtour == 1) {
    $sCriteriaString = $sCriteriaString . ",9";
  }
  if ($theatre == 1) {
    $sCriteriaString = $sCriteriaString . ",10";
  }
  if ($fayre == 1) {
    $sCriteriaString = $sCriteriaString . ",11";
  }
  if ($political == 1) {
    $sCriteriaString = $sCriteriaString . ",12";
  }
  if ($music == 1) {
    $sCriteriaString = $sCriteriaString . ",13";
  }
  if ($tempstruct == 1) {
    $sCriteriaString = $sCriteriaString . ",14";
  }
  if ($public == 1) {
    $sCriteriaString = $sCriteriaString . ",15";
  }
  if ($road == 1) {
    $sCriteriaString = $sCriteriaString . ",16";
  }
  if ($intbar == 1) {
    $sCriteriaString = $sCriteriaString . ",17";
  }

  //close the bracket
  $sCriteriaString = $sCriteriaString . ")";

  //Create a new event
  //If there is a date, insert this with the record.
  //Code to create a new event
  if ($final_event_date != "NULL") {
    $create_event_SQL = "INSERT INTO EVENT (vEventName, dEventDate, vTempFlagSessionID) VALUES
(\"".$eventname."\", \"".$final_event_date."\", \"".$PHPSESSID."\")";
  } else {
    $create_event_SQL = "INSERT INTO EVENT (vEventName, vTempFlagSessionID) VALUES
(\"".$eventname."\", \"".$PHPSESSID."\")";
  }
  mysql_query($create_event_SQL);

  //Create the entry into the EVENTtoUSER table - using the userId in the session variables
  //To do this, select the new eventId out of the event table with the new flag = sessionid, then
reset the flag to 0 later.

  //Get the new event ID
  $get_new_eventid_SQL = "SELECT ipkEventID FROM EVENT WHERE vTempFlagSessionID =
\"".$PHPSESSID."\"";

  $result_get_new_eventid = mysql_query($get_new_eventid_SQL);
  $resultarray_get_new_eventid = mysql_fetch_array($result_get_new_eventid);
  $new_eventid = $resultarray_get_new_eventid["ipkEventID"];

  //Update the session variable
  $_SESSION["session_EventId"] = $new_eventid;

  //Create record in the EVENTtoUSER table.
  $create_EventUser_Link_SQL = "INSERT INTO EVENTtoUSER (iEventID, iUserID) VALUES
(".$_SESSION["session_EventId"].", ".$_SESSION["session_UserId"].")";

  mysql_query($create_EventUser_Link_SQL);

  //Now we have the event id, can reset the temp variable
  $reset_event_table_flag_SQL = "UPDATE EVENT SET vTempFlagSessionID = 0 WHERE ipkEventID =
".$_SESSION["session_EventId"]."";

  mysql_query($reset_event_table_flag_SQL);

  //DEALING WITH GUIDELINES

  //Find out how many sections will be required for this event
  $query_get_sections = "SELECT distinct s.ipkSectionID, s.vSectionName FROM CRITERIA c,
GUIDELINEStoCRITERIA gtc,  GUIDELINES g, SECTIONS s WHERE c.ipkCriteriaID = gtc.iCriteriaID AND
gtc.iGuidelineID = g.ipkGuidelineID AND c.ipkCriteriaID IN ".$sCriteriaString. " AND g.siSectionID =
s.ipkSectionID AND ipkSectionID is not null ORDER BY s.ipkSectionID";

  $result_sectionlist = mysql_query($query_get_sections);
  $num_sections = mysql_num_rows($result_sectionlist);

  //For each section add the relevant records
  for ($i=0; $i < $num_sections; $i++)
  {
    $sectionrow = mysql_fetch_array($result_sectionlist);

    $this_section_name = $sectionrow["vSectionName"];
    $this_section_id = $sectionrow["ipkSectionID"];

    //Get next available set number for guidelines
    $query_get_next_SetID = "SELECT max(iGuidelineSetID) as MaxSetID from EVENTtoGUIDELINESET";
    $result_next_SetID = mysql_query($query_get_next_SetID);
    $next_SetID_row = mysql_fetch_array($result_next_SetID);
    $next_SetIDtoUse = $next_SetID_row["MaxSetID"]+1;

    //RESOURCE PART
    $query_get_next_ResourceSetID = "SELECT max(iResourceSetID) as MaxResourceSetID from
EVENTtoRESOURCESET";
    $result_next_ResourceSetID = mysql_query($query_get_next_ResourceSetID);
    $next_ResourceSetID_row = mysql_fetch_array($result_next_ResourceSetID);
    $next_ResourceSetIDtoUse = $next_ResourceSetID_row["MaxResourceSetID"]+1;

        $create_plan_section_SQL = "INSERT INTO EVENTPLANSECTION (iEventID, iPlanningSectionID,
iGuidelineSetID) VALUES
(".$_SESSION["session_EventId"].",".$this_section_id.",".$next_SetIDtoUse.")";

    mysql_query($create_plan_section_SQL);

    //Get GUIDELINES that go in the section
    $query_get_items_for_section = "SELECT distinct g.ipkGuidelineID, vGuidelineText FROM
GUIDELINES g, GUIDELINEStoCRITERIA gtc WHERE gtc.iGuidelineID = g.ipkGuidelineID AND g.siSectionID =
".$this_section_id." AND iCriteriaID in ".$sCriteriaString." ORDER BY 1";

    $result_itemlist = mysql_query($query_get_items_for_section);
    $num_items = mysql_num_rows($result_itemlist);

    for ($j=0; $j < $num_items; $j++)
    {
      $sectionitem_row = mysql_fetch_array($result_itemlist);

      $create_event_guidelineSet_link_SQL = "INSERT INTO EVENTtoGUIDELINESET (iGuidelineSetID,
iGuidelineID) VALUES (".$next_SetIDtoUse.",".$sectionitem_row["ipkGuidelineID"].")";

      mysql_query($create_event_guidelineSet_link_SQL);
    }

    //Get RESOURCES that go in the section
    $query_get_resources_for_section = "SELECT distinct r.ipkResourceID, vResourceText FROM
RESOURCES r, RESOURCEStoCRITERIA rtc WHERE rtc.iResourceID = r.ipkResourceID AND r.siSectionID =
".$this_section_id." AND iCriteriaID in ".$sCriteriaString." ORDER BY 1";

    $result_resourcelist = mysql_query($query_get_resources_for_section);
    $num_resources = mysql_num_rows($result_resourcelist);

    if ($num_resources > 0) {
      for ($k=0; $k < $num_resources; $k++)
      {
        $sectionresource_row = mysql_fetch_array($result_resourcelist);

        //For each resource, put a record in the EVENTtoRESOURCESET table.
        $create_event_ResourceSet_link_SQL = "INSERT INTO EVENTtoRESOURCESET (iResourceSetID,
iResourceID) VALUES (".$next_ResourceSetIDtoUse.",".$sectionresource_row["ipkResourceID"].")";

        mysql_query($create_event_ResourceSet_link_SQL);
```

```
        //Update EPS record with ResourceSetID
        $update_EPS_record_with_ResourceId_SQL = "UPDATE EVENTPLANSECTION SET iResourceSetID =
".$next_ResourceSetIDtoUse. " WHERE iEventID = ".$_SESSION["session_EventId"]." AND
iPlanningSectionID = ".$this_section_id;

        mysql_query($update_EPS_record_with_ResourceId_SQL);

      }
    }
  }

  //take the user back to their planning section (with the note now updated)
  header("Location:
../pages/planevent.php?PHPSESSID=".$PHPSESSID."&EventId=".$_SESSION["session_EventId"]);

?>

</body>
</html>
styles.css
H2 { color: green; font-size: 37px; font-family: impact }
P.first { text-indent: 1cm; background: yellow; font-family: courier }
font.red { color: red; font-size: 10pt; font-family: Arial, Helvetica, sans-serif }
font.gap { color: black; font-size: 12pt; font-family: Arial, Helvetica, sans-serif line-height:
20px}
font.black { color: black; font-size: 10pt; font-family: Arial, Helvetica, sans-serif }
font.big { color: black; font-size: 14pt; font-family: Arial, Helvetica, sans-serif }
font.bigwhitetitle { color: white; font-size: 26pt; font-family: Arial, Helvetica, sans-serif }
font.white { color: white; font-size: 12pt; font-family: Arial, Helvetica, sans-serif }
font.link { color: blue; font-size: 10pt; font-family: Arial, Helvetica, sans-serif; border-bottom:
solid 1px blue}
font.small { color: black; font-size: 10pt; font-family: Arial, Helvetica, sans-serif }
font.smallred { color: red; font-size: 10pt; font-family: Arial, Helvetica, sans-serif }
font.smalltitle { color: black; font-size: 10pt; font-family: Arial, Helvetica, sans-serif}
font.medwhitetitle { color: white; font-size: 14pt; font-family: Arial, Helvetica, sans-serif }
font.msgtitle { color: black; font-size: 12pt; font-family: Arial, Helvetica, sans-serif }
font.breadcrumb { color: brown; font-size: 12pt; font-family: Arial, Helvetica, sans-serif }

:link { color: blue; font-size: 10pt; font-family: Arial, Helvetica, sans-serif }
:visited { color: blue; font-size: 10pt; font-family: Arial, Helvetica, sans-serif }
```

toolbar.php

```
<?php
?>
<table>
  <tr>
    <td width="107">
      <?php
        echo "<a href=\"../pages/homepage.php?PHPSESSID=".$PHPSESSID."\">";
      ?>
      <img border="0" src="../buttons/toolHome.jpg" alt="home page" width="129" height="34"></a>
    </td>
  </tr>
  <tr>
    <td width="107">
      <img border="0" src="../buttons/toolRes.jpg" width="129" alt="Click for resources"
height="34">
    </td>
  </tr>
  <tr>
    <td width="107">
      <img border="0" src="../buttons/toolTrain.jpg" width="129" alt="Click for training"
height="34">
    </td>
  </tr>
   <tr>
    <td width="107">
      <?php
        echo "<a href=\"../pages/help.php?PHPSESSID=".$PHPSESSID."\">";
      ?>
      <img border="0" src="../buttons/toolHelp.jpg" width="129" alt="Click for help"
height="34"></a>
    </td>
  </tr>
  <tr>
    <td width="107">
      <img border="0" src="../buttons/toolOpt.jpg" width="129" alt="Go to your options"
height="34">
    </td>
  </tr>
  <tr>
    <td width="107">
      <?php
        echo "<a href=\"../functions/logout.php\">";
      ?>
      <img border="0" src="../buttons/toolLogout.jpg" width="129" alt="Logout" height="34"></a>
    </td>
  </tr>
</table>
```

undoChoice.php

```
<?php
  include ("../functions/checklogin.php");
?>
<html>
<head>
  <title>Setting up the event plan</title>
  <?php
    redirect_to_login($redirect);
  ?>

  <LINK REL=StyleSheet HREF="../functions/styles.css" TYPE="text/css">

</head>
<body>
<h1>Sort out the notes updates</h1>
<?php
  connect_to_db();

  //Get in those variables that have been posted
  $buttonpress_UndoChoice = $_POST["undoChoice"];
  $EPSId = $_POST["EPSId"];
  $ChosenQuoteSetID = $_POST["ChosenQuoteSetID"];

  //Check that vars are passed correctly
  echo "<br>buttonpress_UndoChoice is: ".$buttonpress_UndoChoice;
  echo "<br>EPSId is: ".$EPSId;
  echo "<br>ChosenQuoteSetID is: ".$ChosenQuoteSetID;

  if ($buttonpress_UndoChoice) {

    //Delete records from EVENTtoCHOSENQUOTESET
    $Delete_chosen_records_links_SQL = "DELETE FROM EVENTtoCHOSENQUOTESET WHERE iChosenQuoteSetID =
".$ChosenQuoteSetID;
    echo "<br>Delete links sql is: ".$Delete_chosen_records_links_SQL;
    mysql_query($Delete_chosen_records_links_SQL);

    //Update the record in EVENTPLANSECTION
    $Update_EPS_record_SQL = "UPDATE EVENTPLANSECTION SET iChosenQuoteSetID = 0 WHERE
ipkEventPlanSectionID = ".$EPSId;
    echo "<br>Update SQL is: ".$Update_EPS_record_SQL;
    mysql_query($Update_EPS_record_SQL);
  }
  //Redirect - DO THIS AS A FUNC???
  header("Location: ../pages/plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId."#QUOTES");
?>
```

```
</body>
</html>
```

updateNotes.php

```
<?php
  include ("../functions/checklogin.php");
?>

<html>
<head>
  <title>Setting up the event plan</title>
  <?php
    redirect_to_login($redirect);
  ?>

  <LINK REL=StyleSheet HREF="../functions/styles.css" TYPE="text/css">
</head>
<body>
<h1>Sort out the notes updates</h1>
<?php
  connect_to_db();

  //Get in those variables that have been posted
  $notesText = addslashes($_POST["notesText"]);
  $EPSId = $_POST["EPSId"];
  $NotesID = $_POST["NotesID"];

  $buttonpress_UpdateNotes = $_POST["UpdateNotes"];
  $buttonpress_DeleteNotes = $_POST["DeleteNotes"];

  echo "notestext is: ".$notesText;
  echo "<br>EPSId is: ".$EPSId;
  echo "<br>NotesID is: ".$NotesID;

  if ($buttonpress_UpdateNotes) {

    //if a note already exists and this is an update, update it, else add new record and update EPS
table also
    if ($NotesID != 0) {
      $Update_notes_record_SQL = "UPDATE NOTES SET tNotesText = \"".$notesText."\" WHERE ipkNotesID
= ".$NotesID;
      echo "<br>Update query is: ".$Update_notes_record_SQL;
      mysql_query($Update_notes_record_SQL);
    } else {
      $Insert_new_note_SQL = "INSERT INTO NOTES (tNotesText, vTempFlagSessionID) VALUES
(\"".$notesText."\", \"".$PHPSESSID."\")";
      echo "<br>Insert query is: ".$Insert_new_note_SQL;
      mysql_query($Insert_new_note_SQL);

      //Get the new notesID number
      $Get_new_notesId_SQL = "SELECT ipkNotesID FROM NOTES WHERE vTempFlagSessionID =
\"".$PHPSESSID."\"";
      $result_get_new_noteId = mysql_query($Get_new_notesId_SQL);
      $resultarray_Get_new_notesId = mysql_fetch_array($result_get_new_noteId);
      $new_noteId = $resultarray_Get_new_notesId["ipkNotesID"];
      echo "<br>New note is is: ".$new_noteId;

      //Update EPS RECORD
      $Update_EPS_record_SQL = "UPDATE EVENTPLANSECTION SET iNotesID = ".$new_noteId." WHERE
ipkEventPlanSectionID = ".$EPSId;
      echo "Update SQL is: ".$Update_EPS_record_SQL;
      mysql_query($Update_EPS_record_SQL);

      //Reset the temp flag
      $reset_notes_table_flag_SQL = "UPDATE NOTES SET vTempFlagSessionID = 0 WHERE ipkNotesID =
".$new_noteId;
      echo "The reset SQL is: ".$reset_notes_table_flag_SQL;
      mysql_query($reset_notes_table_flag_SQL);
    }

    //take the user back to their planning section (with the note now updated)
    header("Location: ../pages/plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId."#PN");

  } elseif ($buttonpress_DeleteNotes) {

    //If a notes id actually exists....
    if ($NotesID != 0) {

      //Delete record from Notes
      $Delete_notes_record = "DELETE FROM NOTES WHERE ipkNotesID = ".$NotesID;
      echo "Delete note query is: ".$Delete_notes_record;
      mysql_query($Delete_notes_record);

      //Reset EPS record
      $Update_EPS_record_SQL = "UPDATE EVENTPLANSECTION SET iNotesID = 0 WHERE
ipkEventPlanSectionID = ".$EPSId;
      echo "Update SQL is: ".$Update_EPS_record_SQL;
      mysql_query($Update_EPS_record_SQL);
    }
    //take the user back to their planning section (with the note now updated)
    header("Location: ../pages/plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId);
  } else {
    //This state should never be reached - but just in case....
    //take the user back to their planning section (with the note now updated)
    header("Location: ../pages/plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId);
  }
?>
```

updateQuotes.php

```
<?php
  include ("../functions/checklogin.php");
?>
<html>
<head>
  <title>Setting up the event plan</title>
  <?php
    redirect_to_login($redirect);
  ?>

  <LINK REL=StyleSheet HREF="../functions/styles.css" TYPE="text/css">
</head>
<body>
<h1>Sort out the quotes updates</h1>
<?php

  connect_to_db();
  //Get in those variables that have been posted (once - i.e. not the dynamic ones)
  $num_quotes = $_POST["num_quotes"];
  $EPSId .= $_POST["EPSId"];
  $QuotesSetID = $_POST["QuotesSetID"];

  //Have any buttons been pressed?
  $buttonpress_SaveNewQuote = $_POST["saveNewQuote"];
  $buttonpress_UpdateAllQuotes = $_POST["UpdateQuotes"];
  $buttonpress_ConfirmQuotes = $_POST["ConfirmQuotes"];

  //Do I need to add a new quote?
  if ($buttonpress_SaveNewQuote) {
    echo "<br>I am saving this quote as a new one.";
    //GET _n vars (the new quote data)
    $company_n = $_POST["company_n"];
    $contactName_n = $_POST["contactName_n"];
    $contactNumber_n = $_POST["contactNumber_n"];
    $cost_n = $_POST["cost_n"];
    $quoteNotes_n = addslashes($_POST["quoteNotes_n"]);
```

```
    //Add records into the database
    //If first record, ie. num_rows == 0, then create a new QuoteSet.
    if ($QuotesSetID == 0) {

       //Get next available set number for quotes (adapted from setupplan.php)
       $query_get_next_SetID = "SELECT max(iQuotesSetID) as MaxSetID from EVENTtoQUOTESET";
       $result_next_SetID = mysql_query($query_get_next_SetID);
       $next_SetID_row = mysql_fetch_array($result_next_SetID);
       $QuotesSetID = $next_SetID_row["MaxSetID"]+1;
       echo "The new quote set id is: ".$QuotesSetID;

       //Update EPS RECORD
       $Update_EPS_record_SQL = "UPDATE EVENTPLANSECTION SET iQuotesSetID = ".$QuotesSetID." WHERE
ipkEventPlanSectionID = ".$EPSId;
       echo "Update SQL is: ".$Update_EPS_record_SQL;
       mysql_query($Update_EPS_record_SQL);

    }
       //Insert the new quote
       $Insert_new_quote_SQL = "INSERT INTO QUOTES (vCompany, vContactName, vContactNumber, vCost,
vDetails, vTempFlagSessionID) VALUES (\"".$company_n."\",\"".$contactName_n."\"
,\"".$contactNumber_n."\" , \"".$cost_n."\" , \"".$quoteNotes_n."\" , \"".$PHPSESSID."\")";
       echo "<br>Insert query is: ".$Insert_new_quote_SQL;
       mysql_query($Insert_new_quote_SQL);

       //Get the new quoteID number
       $Get_new_quoteId_SQL = "SELECT ipkQuoteID FROM QUOTES WHERE vTempFlagSessionID =
\"".$PHPSESSID."\"";
       $result_get_new_quoteId = mysql_query($Get_new_quoteId_SQL);
       $resultarray_Get_new_quoteId = mysql_fetch_array($result_get_new_quoteId);
       $new_quoteId = $resultarray_Get_new_quoteId["ipkQuoteID"];
       echo "<br>New quoteId is is: ".$new_quoteId;

       //Reset the temp flag
       $reset_quotes_table_flag_SQL = "UPDATE QUOTES SET vTempFlagSessionID = 0 WHERE ipkQuoteID =
".$new_quoteId;
       echo "The reset SQL is: ".$reset_quotes_table_flag_SQL;
       mysql_query($reset_quotes_table_flag_SQL);

       //Add the link record - into EVENTtoQUOTESET
       $Insert_new_quote_link_SQL = "INSERT INTO EVENTtoQUOTESET (iQuotesSetID, iQuoteID) VALUES
(\"".$QuotesSetID."\", \"".$new_quoteId."\")";
       echo "<br>Insert query is: ".$Insert_new_quote_link_SQL;
       mysql_query($Insert_new_quote_link_SQL);

       //Redirect - as no more script needs to be run through - DO THIS AS A FUNC???
       header("Location:
../pages/plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId."#QUOTES");

  }

  //Does something require deletion?
  $delflag = 0;
  $counter = 0;
  while ( ($counter < $num_quotes) && ($delflag == 0) ) {
    $delete = "delQuote_" . $counter;
    $delete2 = "delQuote_" . $counter;
    $$delete = $HTTP_POST_VARS[$delete2];
    echo "<br>del req var is: ".$$delete;

    //If this is the quote to be deleted, set the delflag (to stop the loop iterating), and get the
quoteID.
    if ($$delete) {
      echo "<br>This quote is to be deleted";
      $delflag = 1;
      //Get the quoteId of the current record
      $quoteID = "quoteID_" . $counter;
      $quoteID2 = "quoteID_" . $counter;
      $$quoteID = $HTTP_POST_VARS[$quoteID2];
      echo "<br>The current quote id is: ".$$quoteID;

      //Delete the record from QUOTES
      $Delete_from_QUOTES_SQL = "DELETE FROM QUOTES WHERE ipkQuoteID = ".$$quoteID."";
      echo "<br>delete Q query is: ".$Delete_from_QUOTES_SQL;
      mysql_query($Delete_from_QUOTES_SQL);

      //Delete the record from EVENTtoQUOTESET
      $Delete_from_EVENTtoQUOTESET_SQL = "DELETE FROM EVENTtoQUOTESET WHERE iQuoteID = ".$$quoteID;
      echo "<br>delete EQS query is: ".$Delete_from_EVENTtoQUOTESET_SQL;
      mysql_query($Delete_from_EVENTtoQUOTESET_SQL);

      //Are there now any quotes left in the quote set? If not, reset the flag in the EPS table
      $Check_QuoteSet_still_valid = "SELECT * FROM EVENTtoQUOTESET WHERE iQuotesSetID =
".$QuotesSetID;
      $Get_number_in_quotesSet = mysql_query($Check_QuoteSet_still_valid);
      $num_quotes_in_set= mysql_num_rows($Get_number_in_quotesSet);
      echo "<br>The num rows is: ".$num_quotes_in_set;

      if ($num_quotes_in_set == 0) {
        //reset the variable in EPS
        $Update_EPS_record_SQL = "UPDATE EVENTPLANSECTION SET iQuotesSetID = 0 WHERE
ipkEventPlanSectionID = ".$EPSId;
        echo "<br>Update SQL is: ".$Update_EPS_record_SQL;
        mysql_query($Update_EPS_record_SQL);
      }

      //Redirect - as no more script needs to be run through - DO THIS AS A FUNC???
      header("Location:
../pages/plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId."#QUOTES");

    }
    $counter++;
  }

  //Update all the records
  if ($buttonpress_UpdateAllQuotes) {

    echo "<br>I am about to do alot of updating!";

    //Declare arrays - for all the values to be stored in
    $companyarray = array();
    $contactNamearray = array();
    $contactNumberarray = array();
    $costarray = array();
    $detailsarray = array();
    $quoteIdarray = array();

    $counter = 0;
    //If doing an update, will need this information
    //For the number of quotes, get all the values and put them into arrays - 1 quote at a time
    while ( $counter < $num_quotes ) {

      //Get company values
      $company = "company_" . $counter;
      $company2 = "company_" . $counter;
      $$company = $HTTP_POST_VARS[$company2];
      echo "<br>Var is: ".$$company;
      array_push ($companyarray, $$company);

      //Get contact name values
      $contactName = "contactName_" . $counter;
      $contactName2 = "contactName_" . $counter;
      $$contactName = $HTTP_POST_VARS[$contactName2];
      echo "<br>Var is: ".$$contactName;
      array_push ($contactNamearray, $$contactName);

      //Get contactNumber values
      $contactNumber = "contactNumber_" . $counter;
      $contactNumber2 = "contactNumber_" . $counter;
      $$contactNumber = $HTTP_POST_VARS[$contactNumber2];
      echo "<br>Var is: ".$$contactNumber;
      array_push ($contactNumberarray, $$contactNumber);

      //Get cost values
```

```
        $cost = "cost_" . $counter;
        $cost2 = "cost_" . $counter;
        $$cost = $HTTP_POST_VARS[$cost2];
        echo "<br>Var is: ".$$cost;
        array_push ($costarray, $$cost);

        //Get extra details values
        $details = "quoteNotes_" . $counter;
        $details2 = "quoteNotes_" . $counter;
        $$details = $HTTP_POST_VARS[$details2];
        echo "<br>Var is: ".$$details;
        array_push ($detailsarray, $$details);

        //Get quoteId values
        //Get the quoteId of the current record
        $quoteID = "quoteID_" . $counter;
        $quoteID2 = "quoteID_" . $counter;
        $$quoteID = $HTTP_POST_VARS[$quoteID2];
        echo "<br>The current quote id is: ".$$quoteID;
        array_push ($quoteIdarray, $$quoteID);
        //Update this quote
        $Update_this_quote_SQL = "UPDATE  QUOTES SET vCompany = \"".$$company."\", vContactName =
\"".$$contactName."\", vContactNumber = \"".$$contactNumber."\", vCost = \"".$$cost."\", vDetails =
\"".$$details."\" WHERE ipkQuoteID = ".$$quoteID;
        echo "<br>The update SQL = ".$Update_this_quote_SQL;
        mysql_query($Update_this_quote_SQL);

        $counter++;
     }

     //Redirect - as no more script needs to be run through - DO THIS AS A FUNC???
     header("Location: ../pages/plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId."#QUOTES");

   }

   if ($buttonpress_ConfirmQuotes){
   echo "<br>Confirm button press: ".$buttonpress_ConfirmQuotes;

     //loop through all the quotes, and grab the quoteId of any classed as "confirmed".
     $chosenquotesarray = array();
     $counter = 0;

     while ( $counter < $num_quotes ) {
        $chosen = "chosen_" . $counter;
        $chosen2 = "chosen_" . $counter;
        $$chosen = $HTTP_POST_VARS[$chosen2];

        //Get an array of quoteIds which have been confirmed
        if ($$chosen == 1) {
          $quoteID = "quoteID_" . $counter;
          $quoteID2 = "quoteID_" . $counter;
          $$quoteID = $HTTP_POST_VARS[$quoteID2];
          //echo "<br>The chosen quote id is: ".$$quoteID;
          array_push ($chosenquotesarray, $$quoteID);
        }

        $counter++;
     }

     if (count($chosenquotesarray) > 0) {
        echo "<br>Something has been chosen!";

        //Set up a new ChosenQuoteSet
        $query_get_next_SetID = "SELECT max(iChosenQuoteSetID) as MaxSetID from
EVENTtoCHOSENQUOTESET";
        $result_next_SetID = mysql_query($query_get_next_SetID);
        $next_SetID_row = mysql_fetch_array($result_next_SetID);
        $ChosenQuotesSetID = $next_SetID_row["MaxSetID"]+1;

        //Update the EVENTPLANSECTION record
        $Update_EPS_record_SQL = "UPDATE EVENTPLANSECTION SET iChosenQuoteSetID =
".$ChosenQuotesSetID." WHERE ipkEventPlanSectionID = ".$EPSId;
        echo "<br>Update SQL is: ".$Update_EPS_record_SQL;
        mysql_query($Update_EPS_record_SQL);

        foreach ($chosenquotesarray as $value) {
          //Add the link record - into EVENTtoQUOTESET
          $Insert_new_chosenquote_link_SQL = "INSERT INTO EVENTtoCHOSENQUOTESET (iChosenQuoteSetID,
iQuoteID) VALUES (\"".$ChosenQuotesSetID."\", \"".$value."\")";
          echo "<br>Insert query is: ".$Insert_new_chosenquote_link_SQL;
          mysql_query($Insert_new_chosenquote_link_SQL);
        }
     }

   //Redirect - as no more script needs to be run through - DO THIS AS A FUNC???
   header("Location: ../pages/plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId);

   }
?>

</body>
</html>
```

Web Pages

Emailforhelp.php

```
<?php
  include ("../functions/checklogin.php");
?>

<html>

<head>
  <title>Bath Student Events</title>

  <?php
    redirect_to_login($redirect);
  ?>


  <LINK REL=StyleSheet HREF="http://www.bath.ac.uk/~ma0car/project/functions/styles.css"
TYPE="text/css">

</head>

<body>

  <?php

  global $str_message;

  include ("../functions/header.php");
  //echo "<br>";
  $str_message = "Welcome to the help pages";
  include ("../functions/messagebar.php");

  //Get the relevant data ready to be used in this form
  connect_to_db();

  //Get User forename, surname and email address
  $Get_User_details_SQL = "SELECT FORENAME, SURNAME, EMAIL FROM DBUSERS WHERE USERID =
".$_SESSION["session_UserId"];
  $Get_User_details_result = mysql_query($Get_User_details_SQL);
  $these_user_details=mysql_fetch_array($Get_User_details_result);

  $Name_of_user = $these_user_details["FORENAME"]." ".$these_user_details["SURNAME"];
  $Email_add_of_user = $these_user_details["EMAIL"];
```

```
   //Get the event name and date
   $Get_event_details_SQL = "SELECT vEventName, dEventDate FROM EVENT WHERE ipkEventID =
".$_SESSION["session_EventId"];
   $Get_event_details_result = mysql_query($Get_event_details_SQL);
   $these_event_details=mysql_fetch_array($Get_event_details_result);

   $Name_of_event = $these_event_details["vEventName"];
   $Date_of_event = date("dS F Y", strtotime($these_event_details["dEventDate"]));

   ?>

   <table border="0" bordercolor = "black" cellpadding="2" cellspacing="0" width="656"
align="center">
      <tr>
         <td width="107" valign="top">
            <?php
               include '../functions/toolbar.php';
            ?>
         </td>

         <td width ="549" valign="top">
            <table>
               <tr width ="549">
                  <td align = "left">
                     <?php
                        //Set up the relevant breadcrumbs
                        echo "<a href=\"../pages/homepage.php?PHPSESSID=".$PHPSESSID."\">";
                        echo "<font class = \"breadcrumb\"><strong>Home</font></a>";

                        echo " > ";
                        echo "<a href=\"../pages/help.php?PHPSESSID=".$PHPSESSID."\">";
                        echo "<font class = \"breadcrumb\"><strong>Help</font></a>";

                        echo " > ";
                        echo "<a href=\"../pages/emailforhelp.php?PHPSESSID=".$PHPSESSID."\">";
                        echo "<font class = \"breadcrumb\"><strong>Email a question to
Charlie</font></a>";

                     ?>
                  </td>
               </tr>

               <tr width ="540">
                  <td width ="540" height="300" valign="top">

                     <table border="2" cellpadding="0" cellspacing="2" style="border-collapse: collapse
overflow: scroll" bordercolor="#111111" width="100%" height="300" valign="top">
                        <tr>
                           <td valign="top">

                              <form name="input" action="../pages/sendemail.php" method="post">
                              <table cellpadding="0" cellspacing="2">
                                 <tr>
                                 <td valign="top" colspan="2">
                                    <font class = "big">
                                       Compose your message:
                                    </font>
                                 </td>
                                 </tr>

                                 <tr>
                                 <td>
                                    <font class = "black">
                                    This question is regarding (Event Name):
                                 </td>

                                 <?php
                                 echo "<td align=\"left\" valign=\"center\"><input type=\"text\"
name=\"event_name\" value = \"".$Name_of_event."\" align=\"center\" maxlength=\"25\" size=\"25\">";
                                 echo "</td>";
                                 ?>

                                 <tr>
                                 <td>
                                    <font class = "black">
                                    Due to be held on:
                                 </td>
                                 <?php
                                 echo "<td align=\"left\" valign=\"center\"><input type=\"text\"
name=\"event_date\" value=\"".$Date_of_event."\" align=\"left\" maxlength=\"25\" size=\"25\">";
                                 echo "</td>";
                                 ?>

                                 </tr>

                                 <tr>
                                    <td> 
                                    </td>
                                 </tr>
                                 <tr>
                                 <td>
                                    <font class = "black">
                                    Enter your question below:
                                 </td>
                                 </tr>
                                 <tr>
                                 <td colspan = 2>
                                    <TEXTAREA NAME="question" COLS=40 ROWS=6 align="left" valign="top">
                                    </TEXTAREA>
                                 </td>
                                 </tr>
                                 <tr>
                                    <td> 
                                    </td>
                                 </tr>
                                 <tr>
                                 <td>
                                    <font class = "black">
                                    Senders name:
                                    <?php
                                       echo $Name_of_user;
                                    ?>
                                 </td>
                                 </tr>
                                 <tr>
                                 <td>
                                    <font class = "black">
                                    Senders email address:
                                    <?php
                                       echo $Email_add_of_user;
                                    ?>
                                 </td>
                                 </tr>
                                 <tr>
                                 <td colspan = 2 align="center">
                                    <font class = "red">(If these details are not correct, you can change
them in "My Options".)
                                 </td>
                                 </tr>
                                 <tr>
                                    <td> 
                                    </td>
                                 </tr>
                                 <tr>
                                 <td colspan=2 align="center">
                                    <input type="submit" value="Send Email" align ="center">
                                 </td>
                                 </tr>
                                 <tr>
                                    <td> 
                                    </td>
                                 </tr>
                              </table>
```

```
                        <?php
                          //Pass other vars
                          echo "<input type=\"hidden\" name=\"PHPSESSID\"
value=\"".$PHPSESSID."\">";
                          echo "<input type=\"hidden\" name=\"Name_of_user\"
value=\"".$Name_of_user."\">";
                          echo "<input type=\"hidden\" name=\"Email_add_of_user\"
value=\"".$Email_add_of_user."\">";
                        ?>

                        </form>
                      </td>
                    </tr>
                  </table>
                </td>
              </tr>
            </table>
          </td>
      </tr>
   </table>
</body>
</html>
```

help.php

```
<?php
  include ("../functions/checklogin.php");
?>
<html>
<head>
  <title>Bath Student Events</title>
  <?php
    redirect_to_login($redirect);
  ?>

  <LINK REL=StyleSheet HREF="http://www.bath.ac.uk/~ma0car/project/functions/styles.css"
TYPE="text/css">
</head>
<body>
  <?php
  global $str_message;

  include ("../functions/header.php");
  //echo "<br>";
  $str_message = "Welcome to the help pages";
  include ("../functions/messagebar.php");

  ?>

  <table border="0" bordercolor = "black" cellpadding="2" cellspacing="0" width="656"
align="center">
    <tr>
      <td width="107" valign="top">
        <?php
          include '../functions/toolbar.php';
        ?>
      </td>
      <td width ="549" valign="top">
        <table>
          <tr width ="549">
            <td align = "left">
              <?php
                //Set up the relevant breadcrumbs
                echo "<a href=\"../pages/homepage.php?PHPSESSID=".$PHPSESSID."\">";
                echo "<font class = \"breadcrumb\"><strong>Home</font></a>";
                echo " > ";
                echo "<a href=\"../pages/help.php?PHPSESSID=".$PHPSESSID."\">";
                echo "<font class = \"breadcrumb\"><strong>Help</font></a>";
               ?>
            </td>
```

```
          </tr>
          <tr width ="540">
            <td width ="540" height="300" valign="top">
              <table border="2" cellpadding="0" cellspacing="2" style="border-collapse: collapse
overflow: scroll" bordercolor="#111111" width="100%" height="300" valign="top">
                <tr>
                  <td valign="top">
                    <table>
                      <tr>
                      <td valign="top">
                        <font class = "big">
                          Help Topics:
                        </font>
                      </td>
                      </tr>
                      <tr>
                      <td>
                        <?php
                        echo "<a
href=\"../pages/emailforhelp.php?PHPSESSID=".$PHPSESSID."\">Email a question to Charlie</a>";
                        ?>
                      </td>
                      </tr>
                    </table>
                  </td>
                </tr>
              </table>
            </td>
          </tr>
        </table>
      </td>
    </tr>
  </table>
</body>
</html>
```

Homepage.php

```
<?php
  include ("../functions/checklogin.php");
  include ("../functions/date_eval.php");
?>
<html>
<head>
  <title>Bath Student Events</title>
  <?php
    redirect_to_login($redirect);
  ?>
  <LINK REL=StyleSheet HREF="http://www.bath.ac.uk/~ma0car/project/functions/styles.css"
TYPE="text/css">
</head>
<body>
  <?php

  global $str_message;
  include ("../functions/header.php");
  $str_message = "Welcome to Bath Student Events";
  include ("../functions/messagebar.php");

  ?>
  <table border="0" bordercolor = "black" cellpadding="2" cellspacing="0" width="656"
align="center">
    <tr>
      <td width="107" valign="top">
        <?php
          include '../functions/toolbar.php';
        ?>
      </td>
      <td width ="549" valign="top">
```

```
          <table>
            <!-- This row sets up the breadcrumbs -->
            <tr width ="549">
               <td align = "left">
                  <?php
                    echo "<a href=\"../pages/homepage.php?PHPSESSID=".$PHPSESSID."\">";
                    echo "<font class = \"breadcrumb\"><strong>Home</font></a>";
                  ?>
               </td>
            </tr>
            <tr width ="540">
               <td width ="540" height="300" valign="top">
                  <table border="2" cellpadding="0" cellspacing="2" style="border-collapse: collapse
overflow: scroll" bordercolor="#111111" width="100%" height="300" valign="top">
                     <tr>
                        <td valign="top">
                           <table>
                              <tr>
                                 <td valign="top">
                                    <font class = "big">
                                       Your Current Events:
                                    </font>
                                 </td>
                              </tr>
                              <?php
                                 //List the events being organised by this user
                                 connect_to_db();
                                 $List_user_events_SQL = "SELECT e.ipkEventID AS EventID, e.vEventName
AS EventName, e.dEventDate as EventDate FROM EVENTtoUSER eu, EVENT e WHERE e.ipkEventID =
eu.iEventID AND iUserID =".$_SESSION["session_UserId"];
                                 $list_events_result =mysql_query($List_user_events_SQL);
                                 $num_events= mysql_num_rows($list_events_result);

                                 for ($i=0; $i < $num_events; $i++)
                                 {
                                    $this_event=mysql_fetch_array($list_events_result);
                                    echo "<tr>";
                                       echo "<td>";
                                       echo "<A class=\"link\"
HREF=\"planevent.php?PHPSESSID=".$PHPSESSID."&EventId=".$this_event["EventID"]."\">".$this_event["Ev
entName"]."</A>";
                                       echo "</td>";

                                       //List days to go
                                       $todayDateforDiff = date("Y-m-d");
                                       $eventdate = $this_event["EventDate"];
                                       echo "<td>";
                                       echo "<font class = small>";
                                       if (!$eventdate) {
                                          echo "Event date not set";
                                       } else {
                                          $DaysToGo = date_diff($eventdate, $todayDateforDiff);
                                          echo Eval_DateDiff($DaysToGo);
                                       }

                                       echo "</font></td>";
                                       echo "</tr>";
                                 }

                                 if ($num_events == 0) {
                                    echo "<tr>";
                                    echo "<td><font class = \"black\">";
                                    echo "You do not have any events active.";
                                    echo "</font></td></tr>";
                                 }
                              ?>
                              <tr>
                                 <td>
                                     
                                 </td>
                              </tr>
                              <tr>
                                 <td>
                                    <font class = "black">
                                       <?php
                                          echo "<A HREF=\"newevent.php?PHPSESSID=".$PHPSESSID."\">Click
here to setup a new event</A>";
                                       ?>
                                    </font>
                                 </td>
                              </tr>
                           </table>
                        </td>
                     </tr>
                  </table>
               </td>
            </tr>
         </table>
      </td>
   </tr>
</table>
</body>
</html>
```

login.php

```
<?php

   //set up error flag
   $show_error = 0;

   //set up redirect flag
   $redirect = 0;

   //Catch the variables if they have been entered.
   $username = $_POST["username"];
   $password = $_POST["password"];

   //If an username and password have been entered, see if they are a valid combination.
   if (($username) AND ($password)) {

      //connect to the database - lifted from prev project
      $db=mysql_connect("midge.bath.ac.uk","ma0car","FkMR722d");
      mysql_select_db("csprojStudentEvents");

      $query = "select UserId from DBUSERS where UserName = \"".$username."\" and UserPassword
=\"".     $password."\"";
      $result =mysql_query($query);
      $num_rows= mysql_num_rows($result);
      $myrow = mysql_fetch_row($result);

      //If the username and password is valid, set up a session and redirect user to their
homepage.
      if($num_rows != 1) {
         $show_error = 1;
      } else {

         //declare variables
         global $session_db;
         global $session_UserId;
         global $PHPSESSID;
         global $session_EventId;

         //register new session
         session_start();
         $PHPSESSID = session_id();
         echo "the sess id is:".$PHPSESSID;
         //register session variables
         session_register("session_UserId");
         session_register("session_EventId");
```

```
          $_SESSION["session_UserId"] = $myrow[0];
          //set EventId to -1 to denote that no event is yet being considered
          $_SESSION["session_EventId"] = -1;
          //Set the re-direct flag
          $redirect = 1;
          //echo "The redirect flag is now set to: ".$redirect;
        }
      }
   ?>

<html>
<head>
   <title>Welcome to Bath Student Events</title>

   <?php
   //if the user HAS entered valid login information logged in, redirect them to the homepage
   if ($redirect == 1)
   {
     header("Location: ../pages/homepage.php?PHPSESSID=".$PHPSESSID."");
   }
   ?>
      <SCRIPT>
        /*This function checks for the presence of data in the username and pasword fields when a
user attempts to login - REFERENCE 1.1*/
        function validate() {
          LUV=loginform.username.value;
          LPV=loginform.password.value;
          if ((LUV=='') || (LPV=='')) {
            alert('Please enter a Username and Password.');
            event.returnValue=false;
          }
        }
      </SCRIPT>


   <LINK REL=StyleSheet HREF="http://www.bath.ac.uk/~ma0car/project/functions/styles.css"
TYPE="text/css">

</head>
<body>
   <?php

   include 'http://www.bath.ac.uk/~ma0car/project/functions/header.php';
   $message = "Welcome";
   include ("http://www.bath.ac.uk/~ma0car/project/functions/messagebar.php");
   ?>

   <table border="0" bordercolor = "black" cellpadding="6" cellspacing="0" width="656"
align="center">

   <tr>
     <td width="262" valign = "top">

        <form name="loginform" action="login.php" method="POST" onsubmit="validate();">
        <table width="40%">
          <tr>
            <td width="35%" colspan = "2" style="border-left-style: solid; border-left-width: 1;
border-right-style: solid; border-right-width: 1; border-top-style: solid; border-top-width: 1">
              <div align="center"><font class="big">
              <strong>Existing Users</strong>
              </font></div></td>
          </tr>
          <tr>
            <td width="40%" colspan = "2" style="border-left-style: solid; border-left-width: 1;
border-right-style: solid; border-right-width: 1">
              <?php
                if ($show_error == 1) {
                  echo "<div align=\"center\"><font class=\"smallred\">";
                  echo "Incorrect Username or Password.<br> Please try again.";
                } else {
                  echo "<div align=\"center\"><font class=\"small\">";
                  echo "Enter your username and password to login:";
                }
              ?>
                 </font></div>
              </td>
            </tr>
            <tr>
              <td width="10%" style="border-left-style: solid; border-left-width: 1">
                <div align="right"><font class="black">Username: </font></div>
              </td>
              <td width="30%" style="border-right-style: solid; border-right-width: 1; border-top-
style: solid; border-top-width: 1">
                <input type="text" name="username" size="30" maxlength="30">
              </td>
            </tr>
            <tr>
              <td width="10%" style="border-left-style: solid; border-left-width: 1">
                <div align="right"><font class="black">Password: </font></div>
              </td>
              <td width="30%" style="border-right-style: solid; border-right-width: 1; border-bottom-
style: solid; border-bottom-width: 1">
                <input type="password" name="password" size="30" maxlength="20">
              </td>
            </tr>
            <tr>
              <td width="40%" colspan = "2" style="border-left-style: solid; border-left-width: 1;
border-right-style: solid; border-right-width:1">
                <div align="center">
                <input type=image value=submit
src="http://www.bath.ac.uk/~ma0car/project/buttons/Login2.jpg" width="61" height="26" border="0"
name="image">
                </div>
              </td>
            </tr>
            <tr>
              <td width="40%" colspan = "2" style="border-left-style: solid; border-left-width: 1;
border-right-style: solid; border-right-width: 1; border-bottom-style: solid; border-bottom-width:
1">
                <div align="right"><font class = "link">
                  Forgotten your password
                </font></div>
              </td>
            </tr>

            <?php
              if ($show_error == 1) {
                echo "<tr>";
                echo "<td width=\"40%\" colspan = \"2\">";
                echo "<div align=\"left\"><font class=\"smallred\">";
                echo "<br>";
                echo "Remember that:";
                echo "<ul>";
                echo "<li>Passwords are case sensitive</li>";
                echo "<li>Check that you do not have CAPS on</li>";
                echo "<li>Click on the link above if you have forgotten your password</li>";
                echo "</ul>";
                echo "</font></div>";
                echo "</td>";
                echo "</tr>";
              }
            ?>

       </table>
       </td>
       <td valign="top">
          <font class = "big"> New to Bath Student Events?</font>
          <font class = "small">
          <br><br>
          Welcome to the unique service provided by the Students' Union to help you run your events
successfully.
          <ul>
            <li>Get Ideas and Advice</li>
            <li>All the resources you need at your fingertips</li>
```

```
        <li>Share information with other organisers</li>
      </ul>

      <strong>Bath University Students can:</strong> </font>
      <font class = "link"> Sign up now </font>
      <br>
      <font class = "small"> (You may need to enter your BUCS username and password) </font>
    </td>
</table>
</body>
</html>
</body>
</html>
```

newevent.php

```
<?php
  include ("../functions/checklogin.php");
  //Set the eventId to -1 as this is a new event
  $_SESSION["session_EventId"] = -1;
?>

<html>
<head>
  <title>Bath Student Events</title>
  <?php
    redirect_to_login($redirect);
  ?>
  <LINK REL=StyleSheet HREF="../functions/styles.css" TYPE="text/css">

</head>
<body>
  <?php
  $error_format_event_name =$_GET['error_format_event_name'];
  $date_error_31 =$_GET['date_error_31'];
  $date_error_feb =$_GET['date_error_feb'];
  $date_error_feb_leap =$_GET['date_error_feb_leap'];

  global $str_message;

  include '../functions/header.php';
  $str_message = "Set up a new event";
  include '../functions/messagebar.php';
  ?>

  <table border="0" bordercolor = "black" cellpadding="2" cellspacing="0" width="656"
align="center">
    <tr>
      <td width="107" valign="top">
        <?php
          include '../functions/toolbar.php';
        ?>
      </td>
      <td width ="549" valign="top">
        <table>
          <tr width ="549">
            <td align = "left">
              <?php
                //Set up the relevant breadcrumbs
                echo "<a href=\"../pages/homepage.php?PHPSESSID=".$PHPSESSID."\">";
                echo "<font class = \"breadcrumb\"><strong>Home</font></a>";

                echo " > ";
                echo "<A HREF=\"../pages/newevent.php?PHPSESSID=".$PHPSESSID."\">";
                echo "<font class = \"breadcrumb\"><strong>Create a new event: Stage
1</font></a>";
              ?>
            </td>
          </tr>
          <tr width ="540">
            <td width ="540" height="300" valign="top">
              <table border="2" cellpadding="2" cellspacing="2" style="border-collapse: collapse
overflow: scroll" bordercolor="#111111" width="100%" height="300" valign="top" bgcolor="FFFFCC">
                <tr>
                  <td valign="top">
                    <?php
                    echo "<form name=\"input\"
action=\"../pages/wizard.php?PHPSESSID=".$PHPSESSID."\" method=\"post\">";
                    ?>
                    <table>
                      <tr>
                        <td valign="top" colspan="3"><font class = "big"> Stage 1 - Event
details</font><br>
                        </td>
                      </tr>
                      <tr>
                        <td valign="top" colspan="3"><font class="small">Please fill in a name
for the event you are planning, that you will know it by, and give the date if you know it yet.
                        </td>
                      </tr>
                      <tr>
                        <td width ="100"><font class="small"><strong>Event name:</strong></td>
                        <td align="left"><input type="text" name="eventname"></td>
                      </tr>

                      <tr>
                        <td colspan="3"><font class="small"><strong>
                        Have you already set a date for your event?</strong>
                        <br>
                        Yes
                        <input type="radio" name="gotdate" value="1"/>
      
                        No
                        <input type="radio" name="gotdate" value="2" checked/>
      
                      </td>
                      </tr>
                      <tr>
                        <td ALIGN=LEFT VALIGN=TOP WIDTH="150"><font class="small"><strong>If
yes, enter the date:</strong></font></td>
                        <td ALIGN=LEFT VALIGN=TOP WIDTH="350"><select name="eventDay"
size="1"><option value="01">1</option><option value="02">2</option><option
value="03">3</option><option value="4">4</option><option value="05">5</option><option
value="06">6</option><option value="07">7</option><option value="08">8</option><option
value="09">9</option><option value="10">10</option><option value="11">11</option><option
value="12">12</option><option value="13">13</option><option value="14">14</option><option
value="15">15</option><option value="16">16</option><option value="17">17</option><option
value="18">18</option><option value="19">19</option><option value="20">20</option><option
value="21">21</option><option value="22">22</option><option value="23">23</option><option
value="24">24</option><option value="25">25</option><option value="26">26</option><option
value="27">27</option><option value="28">28</option><option value="29">29</option><option
value="30">30</option><option value="31">31</option></select>
                        <select name="eventMonth" size="1"><option
value="01">January</option><option value="02">February</option><option
value="03">March</option><option value="04">April</option><option value="05">May</option><option
value="06">June</option><option value="07">July</option><option value="08">August</option><option
value="09">September</option><option value="10">October</option><option
value="11">November</option><option value="12">December</option></select>

                        <select name="eventYear" size="1"><option
value="2004">2004</option><option value="2005">2005</option></select>
                        </td>
                      </tr>
                      <tr>
                        <td colspan="3" align="left">
                          <font class="smallred">
                        <?php

                          if  ($error_format_event_name == 1 ) {
                              echo "The event name must be between 1 and 30 characters long
containing only numbers and letters (and spaces).<br>";
                          }
```

```
                              if ($date_error_31 == 1) {
                                echo "The month you entered does not have 31 days.<br>";
                              }

                              if ( ($date_error_feb == 1) || ($date_error_feb_leap == 1) ) {
                                echo "The February date you entered was not valid.";
                              //This can go in anyway - doesnt matter if test has been put in. But
if there is no text, it will keep the table format.
                              } else {
                                echo " ";
                              }
                            ?>

                            </font></td>
                          </tr>

                          <?php
                          //Pass sessionID
                          echo "<input type=\"hidden\" name=\"PHPSESSID\"
value=\"".$PHPSESSID."\">";

                          ?>
                          <tr>
                            <td colspan="3" align = "right"><input type="submit"
value="Next"/></td>
                          </tr>

                        </table>
                      </form>

                      </td>
                    </tr>
                  </table>
                </td>
              </tr>
            </table>
          </td>
      </tr>
  </table>
</body>
</html>
```

planevent.php

```
<?php
  include ("../functions/checklogin.php");
  //Set the event session ids
  $_SESSION["session_EventId"] = $_GET['EventId'];
  connect_to_db();

  //Set the session event name
  $Get_event_name_SQL = "SELECT vEventName FROM EVENT WHERE ipkEventID =
".$_SESSION["session_EventId"];
  $Get_event_name = mysql_query($Get_event_name_SQL);
  $event_name = mysql_fetch_array($Get_event_name);

  $_SESSION["session_EventName"] = $event_name["vEventName"];
?>
<html>

<head>
  <title>Bath Student Events</title>
  <?php
    redirect_to_login($redirect);
  ?>

  <LINK REL=StyleSheet HREF="../functions/styles.css" TYPE="text/css">
</head>
<body>
  <?php
    global $str_message;
    include ("../functions/header.php");
    $str_message = "Event: ".$_SESSION["session_EventName"];
    include ("../functions/messagebar.php");
  ?>
  <table border="0" bordercolor = "black" cellpadding="2" cellspacing="0" width="656"
align="center">
    <tr>
      <td width="107" valign="top">
        <?php
          include '../functions/toolbar.php';
        ?>
      </td>
      <td width ="549" valign="top">
        <table>
          <tr width ="549">
            <td align = "left">
              <?php
                //Set up the relevant breadcrumbs
                echo "<a href=\"../pages/homepage.php?PHPSESSID=".$PHPSESSID."\">";
                echo "<font class = \"breadcrumb\"><strong>Home</font></a>";
                echo " > ";
                echo "<a
HREF=\"../pages/planevent.php?PHPSESSID=".$PHPSESSID."&EventId=".$_SESSION["session_EventId"]."\">";
                echo "<font class =
\"breadcrumb\"><strong>".$_SESSION["session_EventName"]."</font></a>";
              ?>
            </td>
          </tr>

          <tr width ="540">
            <td width ="540" height="300" valign="top">

              <table border="2" cellpadding="0" cellspacing="2" style="border-collapse: collapse
overflow: scroll" bordercolor="#111111" width="100%" height="300" valign="top">
                <tr>
                  <td valign="top">

                    <table>
                      <tr>
                        <td valign="top">
                          <font class = "big">
                            Plan your event
                          </font>
                          <font class = "black">
                            Click on each section for further information
                          </font>
                        </td>
                      </tr>

                      <?php

                        //List the planning refs for this event



                        $List_event_sections_SQL = "SELECT distinct eps.iPlanningSectionID,
s.vSectionName as SectionName, eps.ipkEventPlanSectionID as EPSId FROM EVENTPLANSECTION eps,
SECTIONS s WHERE s.ipkSectionID = eps.iPlanningSectionID AND eps.iEventID
=".$_SESSION["session_EventId"];

                        //echo $List_event_sections_SQL;

                        $list_event_sections =mysql_query($List_event_sections_SQL);
                        $num_sections= mysql_num_rows($list_event_sections);

                        for ($i=0; $i < $num_sections; $i++)
                        {
                          $this_section=mysql_fetch_array($list_event_sections);
                          echo "<tr>";
                            echo "<td>";
```

```
                                    echo "<A class=\"link\"
HREF=\"plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$this_section["EPSId"]."\">".$this_section["
SectionName"]."</A>";
                                    echo "</td>";
                                  echo "</tr>";
                              ?>
                              <tr>
                                 <td>
                                    
                                 </td>
                              </tr>
                            </table>
                          </td>
                        </tr>
                      </table>
                    </td>
                  </tr>
                </table>
              </td>

    </tr>
  </table>
</body>
</html>
```

plansection.php

```
<?php
  include ("../functions/checklogin.php");
  connect_to_db();
  //Get the variable of the planning section id ref
  $EPSId = $_GET['EPSId'];

  //Get the variable to see if a request has been made to add notes or quotes
  $AddNotes = $_GET['AddNotes'];
  $AddQuote = $_GET['AddQuote'];

  //SET  UP THE KEY VARIABLES

  //Get section name for title of area
  $Get_section_name_SQL = "SELECT s.vSectionName as SectionName, eps.iGuidelineSetID as
GuidelineSetID, eps.iResourceSetID as ResourceSetID, eps.iPostChoiceToDoSetID as
PostChoiceToDoSetID, eps.iNotesID as NotesID, eps.iQuotesSetID as QuotesSetID, eps.iChosenQuoteSetID
as ChosenQuoteSetID FROM EVENTPLANSECTION eps, SECTIONS s WHERE s.ipkSectionID =
eps.iPlanningSectionID AND eps.ipkEventPlanSectionID =".$EPSId;

  $Get_section_name_result =mysql_query($Get_section_name_SQL);
  $section_name_result=mysql_fetch_array($Get_section_name_result);

  //Also got the link ids out of the query above for use here...
  $SectionName = $section_name_result["SectionName"];
  $GuidelineSetID = $section_name_result["GuidelineSetID"];
  $ResourceSetId = $section_name_result["ResourceSetID"];
  $PostChoiceToDoSetID = $section_name_result["PostChoiceToDoSetID"];
  $NotesID =  $section_name_result["NotesID"];
  $QuotesSetID =  $section_name_result["QuotesSetID"];
  $ChosenQuoteSetID = $section_name_result["ChosenQuoteSetID"];

?>

<html>
<head>
  <title>Bath Student Events</title>
  <?php
    redirect_to_login($redirect);
  ?>
  <LINK REL=StyleSheet HREF="../functions/styles.css" TYPE="text/css">
</head>
<body>
  <?php
    global $str_message;

    include ("../functions/header.php");
    //echo "<br>";
    $str_message = "Event: ".$_SESSION["session_EventName"];
    include ("../functions/messagebar.php");

  ?>


  <table border="0" bordercolor = "black" cellpadding="2" cellspacing="0" width="656"
align="center">
    <tr>
      <td width="107" valign="top">
        <?php
          include '../functions/toolbar.php';
        ?>
      </td>

      <td width ="549" valign="top">
        <table>
          <tr width ="549">
            <td align = "left">
              <?php
                //Set up the relevant breadcrumbs
                echo "<a href=\"../pages/homepage.php?PHPSESSID=".$PHPSESSID."\">";
                echo "<font class = \"breadcrumb\"><strong>Home</font></a>";

                echo " > ";
                echo "<a
HREF=\"../pages/planevent.php?PHPSESSID=".$PHPSESSID."&EventId=".$_SESSION["session_EventId"]."\">";
                echo "<font class =
\"breadcrumb\"><strong>".$_SESSION["session_EventName"]."</font></a>";

                echo " > ";
                echo "<A
HREF=\"../pages/plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId."\">";
                echo "<font class = \"breadcrumb\"><strong>".$SectionName."</font></a>";
              ?>
            </td>
          </tr>

          <tr width ="540">
            <td width ="540" height="300" valign="top">

              <table border="2" cellpadding="0" cellspacing="2" style="border-collapse: collapse
overflow: scroll" bordercolor="#111111" width="100%" height="300" valign="top">
              <tr>
              <td valign="top">

                <table>

                  <?php

                    //Print section title
                    echo "<tr>";
                      echo "<td valign=\"top\">";
                      echo "<font class = \"big\">";
                      echo $SectionName;
                      echo "</font></td>";
                    echo "</tr>";

                    if ($ChosenQuoteSetID != 0) {

                      //Set up a totally different page! - for post setup
                      include ("../functions/postchoiceSetup.php");

                    } else {

                    //Print guidelines heading
                    echo "<tr>";
```

```
                        echo "<td bgcolor=\"silver\" valign=\"top\" width=\"500\">";
                        echo "<font class = \"black\"><strong>";
                        echo "Guidelines";
                        echo "</strong></font></td>";
                     echo "</tr>";

                     //List guidelines

                     $Get_section_guidelines_SQL = "SELECT g.vGuidelineText as GuidelineText FROM
EVENTtoGUIDELINESET egs, GUIDELINES g WHERE egs.iGuidelineID = g.ipkGuidelineID AND
egs.iGuidelineSetID=".$GuidelineSetID;

                     $Get_guidelines = mysql_query($Get_section_guidelines_SQL);
                     $num_guidelines= mysql_num_rows($Get_guidelines);

                     echo "<tr>";
                     echo "<td>";
                     echo "<font class = \"black\">";
                     echo "<ul>";

                     for ($i=0; $i < $num_guidelines; $i++)
                     {
                        $this_guideline=mysql_fetch_array($Get_guidelines);
                          echo "<li>";
                          echo $this_guideline["GuidelineText"];
                          echo "</li>";
                     }

                     echo "</ul>";
                     echo "</td>";
                     echo "</tr>";

                     //Print resources heading
                     echo "<tr>";
                        echo "<td bgcolor=\"silver\" valign=\"top\" width=\"500\">";
                        echo "<font class = \"black\"><strong>";
                        echo "Resources";
                        echo "</strong></font></td>";
                     echo "</tr>";

                     //List resources

                     $Get_section_resources_SQL = "SELECT r.vResourceText as ResourceText FROM
EVENTtoRESOURCESET ers, RESOURCES r WHERE ers.iResourceID = r.ipkResourceID AND
ers.iResourceSetID=".$ResourceSetId;

                     $Get_resources = mysql_query($Get_section_resources_SQL);
                     $num_resources= mysql_num_rows($Get_resources);

                     if($num_resources > 0) {
                        echo "<tr>";
                        echo "<td>";
                        echo "<font class = \"black\">";

                        echo "<ul>";
                        for ($i=0; $i < $num_resources; $i++)
                        {
                           $this_resource=mysql_fetch_array($Get_resources);
                             echo "<li>";
                             echo $this_resource["ResourceText"];
                             echo "</li>";
                        }
                        echo "</ul>";

                        echo "</td>";
                        echo "</tr>";
                     } else {
                        echo "<tr>";
                           echo "<td align = \"center\" valign=\"top\" width=\"500\">";
                           echo "<font class = \"black\">";
                           echo "There are currently no resources available for this section";
                           echo "</font></td>";
                        echo "</tr>";
                     }
                     //Print notes heading
                     echo "<tr>";
                        echo "<td bgcolor=\"silver\" valign=\"top\" width=\"500\">";
                        echo "<font class = \"black\"><strong>";
                        echo "<a name = \"PN\"> Personal Notes</a>";
                        echo "</strong></font></td>";
                     echo "</tr>";
                     echo "<tr>";
                     //If notes exist, or if the user wants to add some, setup the form
                     if( ($AddNotes ==1) || ($NotesID != 0) ) {
                        echo "<td>";
                        echo "<form name=\"input\" action=\"../functions/updateNotes.php\"
method=\"post\">";
                        echo "<table>";
                        echo "<tr>";
                           echo "<td rowspan=\"2\">";
                              echo "<TEXTAREA NAME=\"notesText\" COLS=50 ROWS=6>";
                              //if the user has already added notes, put the text in here.
                              if ($NotesID != 0) {
                                 $get_notes_text_SQL = "SELECT tNotesText FROM NOTES WHERE
ipkNotesID = ".$NotesID;

                                 $Get_note_text = mysql_query($get_notes_text_SQL);
                                 $this_note_text=mysql_fetch_array($Get_note_text);
                                 echo stripslashes($this_note_text["tNotesText"]);
                              }
                              echo "</TEXTAREA>";
                           echo "</td>";
                           echo "<td align = \"center\">";
                           echo "<input type=\"submit\" value=\"Update\nNotes\"
name=\"UpdateNotes\" align = \"center\"/>";
                           echo "</td>";
                        echo "</tr>";
                        echo "<tr>";
                           echo "<td align = \"center\">";
                           echo "<input type=\"submit\" value=\"Delete\nNotes\"
name=\"DeleteNotes\" align = \"center\"/>";
                           echo "</td>";
                        echo "</tr>";
                        echo "</table>";
                        //Pass sessionID, sectionID and the notesID
                        echo "<input type=\"hidden\" name=\"PHPSESSID\"
value=\"".$PHPSESSID."\">";
                        echo "<input type=\"hidden\" name=\"EPSId\" value=\"".$EPSId."\">";
                        echo "<input type=\"hidden\" name=\"NotesID\" value=\"".$NotesID."\">";
                        echo "</form>";
                     } else {
                        //If the user has not added any personal notes...
                        echo "<td align = \"center\" valign=\"top\" width=\"500\">";
                        echo "<font class = \"black\">";
                        echo "If you would like to add any personal notes about this section,
click ";
                        echo "<A class=\"link\"
HREF=\"plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId."&AddNotes=1\">Here</A>";
                        echo "</font>";
                     }
                     echo "</td></tr>";
                     //Print quotes heading
                     echo "<tr>";
                        echo "<td bgcolor=\"silver\" valign=\"top\" width=\"500\">";
                        echo "<font class = \"black\"><strong>";
                        echo "<a name = \"QUOTES\">Quotes</a>";
                        echo "</strong></font></td>";
                     echo "</tr>";
                     echo "<tr><td>";
                     $Get_section_quotes_SQL = "SELECT * FROM EVENTtoQUOTESET eqs, QUOTES q WHERE
eqs.iQuoteID = q.ipkQuoteID AND eqs.iQuotesSetID=".$QuotesSetID;
                     $Get_quotes = mysql_query($Get_section_quotes_SQL);
                     $num_quotes= mysql_num_rows($Get_quotes);

                     //If quotes exist, list them
```

```
                        //if($QuotesSetID != 0) {
                        echo "<form name=\"input\" action=\"../functions/updateQuotes.php\"
method=\"POST\">";

                        for ($q=0; $q < $num_quotes; $q++) {

                          //Scoop up the quote details ready to use below
                          $this_quote=mysql_fetch_array($Get_quotes);

                          $thisQuoteID = $this_quote["ipkQuoteID"];
                          $thisCompany = $this_quote["vCompany"];
                          $thisContactName = $this_quote["vContactName"];
                          $thisContactNumber = $this_quote["vContactNumber"];
                          $thisCost = $this_quote["vCost"];
                          $thisDetail = $this_quote["vDetails"];

                          echo "<table>";
                          echo "<tr>";
                            echo "<td width=\"60\" valign=\"center\"><font class =
\"smalltitle\"><strong>";
                              echo "Company:";
                              echo "</td>";
                              echo "<td width=\"60\" valign=\"center\"><input type=\"text\"
name=\"company_".$q."\" value=\"".$thisCompany."\" align=\"center\" maxlength=\"25\" size=\"15\">";
                              echo "</td>";
                              echo "<td width=\"10\" valign=\"center\"><font class =
\"smalltitle\"><strong>";
                              echo "Cost:";
                              echo "<td valign=\"center\"><input type=\"text\"
value=\"".$thisCost."\" name=\"cost_".$q."\" align=\"center\" maxlength=\"10\" size=\"5\">";
                              echo "</td>";
                              echo "<td align = \"center\">";
                                 echo "<input type=\"submit\" value=\"Delete this Quote\" name =
\"delQuote_".$q."\" align = \"center\"/>";
                              echo "</td>";
                            echo "</tr>";
                            echo "<tr>";
                              echo "<td width=\"100\" valign=\"center\"><font class =
\"smalltitle\"><strong>";
                              echo "Contact:";
                              echo "</td>";
                              echo "<td width=\"60\" valign=\"center\"><input type=\"text\"
name=\"contactName_".$q."\" value=\"".$thisContactName."\" align=\"center\" maxlength=\"25\"
size=\"15\">";
                              echo "</td>";
                              echo "<td colspan = \"2\" rowspan=\"2\"><TEXTAREA
NAME=\"quoteNotes_".$q."\" COLS=15 ROWS=3>";
                              echo $thisDetail;
                              echo "</TEXTAREA>";
                              echo "<td valign=\"bottom\" align=\"center\"><font class =
\"smalltitle\"><strong>";
                              echo "Check to confirm:";
                              echo "</td>";
                            echo "</tr>";
                            echo "<tr>";
                              echo "<td width=\"100\" valign=\"center\"><font class =
\"smalltitle\"><strong>";
                              echo "Contact no.:";
                              echo "</td>";
                              echo "<td width=\"60\" valign=\"center\"><input type=\"text\"
value=\"".$thisContactNumber."\" name=\"contactNumber_".$q."\" align=\"center\" maxlength=\"25\"
size=\"15\">";
                              echo "</td>";
                              echo "<td valign=\"top\" align=\"center\"><font class =
\"smalltitle\"><strong>";
                              echo "<input type=\"checkbox\" name=\"chosen_".$q."\" value=\"1\"/>";
                              echo "</td>";
                              echo "</tr>";
                            //Pass each quoteId as a hidden variable
                            echo "<input type=\"hidden\" name=\"quoteID_".$q."\"
value=\"".$thisQuoteID."\">";
                            echo "</table>";
                            echo "<hr>";
                          }

                          //If we are adding a new quote...
                          if ($AddQuote == 1) {
                          echo "<table>";
                            echo "<tr>";
                              echo "<td width=\"60\" valign=\"center\"><font class =
\"smalltitle\"><strong>";
                              echo "Company:";
                              echo "</td>";
                              echo "<td width=\"60\" valign=\"center\"><input type=\"text\"
name=\"company_n\" align=\"center\" maxlength=\"25\" size=\"15\">";
                              echo "</td>";
                              echo "<td width=\"10\" valign=\"center\"><font class =
\"smalltitle\"><strong>";
                              echo "Cost:";
                              echo "<td valign=\"center\"><input type=\"text\" value=\"\"
name=\"cost_n\" align=\"center\" maxlength=\"10\" size=\"5\">";
                              echo "</td>";
                              echo "<td align = \"center\">";
                                 echo "<input type=\"submit\" value=\"Save new Quote\" name =
\"saveNewQuote\" align = \"center\"/>";
                              echo "</td>";
                            echo "</tr>";
                            echo "<tr>";
                              echo "<td width=\"100\" valign=\"center\"><font class =
\"smalltitle\"><strong>";
                              echo "Contact:";
                              echo "</td>";
                              echo "<td width=\"60\" valign=\"center\"><input type=\"text\"
name=\"contactName_n\" align=\"center\" maxlength=\"25\" size=\"15\">";
                              echo "</td>";
                              echo "<td colspan = \"2\" rowspan=\"2\"><TEXTAREA
NAME=\"quoteNotes_n\" COLS=15 ROWS=3>Extra details";
                              echo "</TEXTAREA>";
                            echo "</tr>";
                            echo "<tr>";
                              echo "<td width=\"100\" valign=\"center\"><font class =
\"smalltitle\"><strong>";
                              echo "Contact no.:";
                              echo "</td>";
                              echo "<td width=\"60\" valign=\"center\"><input type=\"text\"
value=\"\" name=\"contactNumber_n\" align=\"center\" maxlength=\"25\" size=\"15\">";
                              echo "</td>";
                            echo "</tr>";
                            echo "</table>";
                            echo "<hr>";
                          }
                          //Pass sessionID, quotesetid and the number of quotes to the form processing
the request
                          echo "<input type=\"hidden\" name=\"PHPSESSID\" value=\"".$PHPSESSID."\">";
                          echo "<input type=\"hidden\" name=\"EPSId\" value=\"".$EPSId."\">";
                          echo "<input type=\"hidden\" name=\"QuotesSetID\"
value=\"".$QuotesSetID."\">";
                          echo "<input type=\"hidden\" name=\"num_quotes\"
value=\"".$num_quotes."\">";
                          //Put in the buttons
                          echo "<table>";
                          echo "<tr>";
                            echo "<td align = \"center\">";
                            echo "<A class=\"link\"
HREF=\"plansection.php?PHPSESSID=".$PHPSESSID."&EPSId=".$EPSId."&AddQuote=1#QUOTES\">";
                            echo "<img border=\"0\" src=\"../buttons/quoteAddNew.jpg\" alt=\"home
page\" width=\"129\" height=\"34\"></a>";
                            echo "</td>";
                            if ($num_quotes > 0 ) {
                              echo "<td align = \"center\">";
                              echo "<input type=\"submit\" value=\"Update all Quotes\" name =
\"UpdateQuotes\" align = \"center\"/>";
                              echo "</td>";
                              echo "<td align = \"center\">";
```

```
                            echo "<input type=\"submit\" value=\"Confirm choice of quotes\" name =
\"ConfirmQuotes\" align = \"center\"/>";
                            echo "</td>";
                          }
                        echo "</tr></table>";
                        echo "</form>";
                        //this is the end of the big else quote
                        }
                      ?>
                      <tr>
                        <td>
                           
                        </td>
                      </tr>
                    </table>
                      </td>
                    </tr>
                  </table>
                </td>
              </tr>
            </table>
          </td>
        </tr>
      </table>
    </body>
    </html>
    sendemail.php
    <?php
      include ("../functions/checklogin.php");
    ?>
    <html>
    <head>
      <title>Bath Student Events</title>
      <?php
        redirect_to_login($redirect);
      ?>
      <LINK REL=StyleSheet HREF="../functions/styles.css" TYPE="text/css">
    </head>
    <body>

      <?php
      global $str_message;
      include ("../functions/header.php");
      //echo "<br>";
      $str_message = "Welcome to the help pages";
      include ("../functions/messagebar.php");
      //Pick up vars for email
      $event_name = $_POST["event_name"];
      $event_date = $_POST["event_date"];
      $question = $_POST["question"];
      $Name_of_user = $_POST["Name_of_user"];
      $Email_add_of_user = $_POST["Email_add_of_user"];
      $adminemail = "ma0car@bath.ac.uk";
      $message = "Events Question via BSE\n\n"."Dear Charlie,\n\nYou have received a question via a
    user on BSE, as detailed below:\n\nSenders name: ".$Name_of_user."\nSenders email address:
    ".$Email_add_of_user."\n\nThe event: ".$event_name."\nThe event date: ".$event_date."\n\nTheir
    Question: ".$question."";
      mail($adminemail, "Events Question via BSE", $message, "From:".$Name_of_user."
    <".$Email_add_of_user.">");
      ?>
      <table border="0" bordercolor = "black" cellpadding="2" cellspacing="0" width="656"
    align="center">
        <tr>
          <td width="107" valign="top">
            <?php
              include '../functions/toolbar.php';
            ?>
          </td>
          <td width ="549" valign="top">
            <table>
              <tr width ="549">
                <td align = "left">
                  NAVIGATION BUTTONS GO IN HERE
                </td>
              </tr>
              <tr width ="540">
                <td width ="540" height="300" valign="top">

                  <table border="2" cellpadding="0" cellspacing="2" style="border-collapse:
    collapse overflow: scroll" bordercolor="#111111" width="100%" height="300" valign="top">
                  <tr>
                    <td valign="top"><font class="black">
                    <?php
                    //Set up and send email

                    if (mail){
                      echo "Your question was emailed successfully.";
                    } else {
                      echo "There was a problem emailing your question to Charlie - please try
    again.";
                    }
                    ?>
                    </font></td>
                  </tr>
                  </table>
                </td>
              </tr>
            </table>
          </td>
        </tr>
      </table>
    </body>
    </html>
```

wizard.php

```
<?php

  include ("../functions/checklogin.php");

  //Lifted from http://uk.php.net/manual/en/function.cal-days-in-month.php
  //Returns the number of days in a given month and year
  function days_in_month($month, $year)
  {
    // calculate number of days in a month
    return $month == 2 ? ($year % 4 ? 28 : ($year % 100 ? 29 : ($year % 400 ? 28 : 29))) : (($month
- 1) % 7 % 2 ? 30 : 31);
  }

  //set up error flags
  $error_format_event_name = 0;
  $date_error_31 = 0;
  $date_error_feb = 0;
  $date_error_feb_leap = 0;

  //set up redirect flag
  $stage1_redirect = 0;

  //Catch the stage 1 variables if they have been entered.
  $eventname = $_POST["eventname"];
  $eventDay = $_POST["eventDay"];
  $eventMonth = $_POST["eventMonth"];
  $eventYear = $_POST["eventYear"];
  $gotdate = $_POST["gotdate"];

  if (!ereg("^[a-zA-Z0-9][a-zA-Z0-9[:space:]]{0,29}$", $eventname)) {
    $error_format_event_name = 1;
    $stage1_redirect = 1;
    //echo "bad format";
  }

  //Set flags to detect wrong dates
```

```
   //Check days with 30 days
   if ( ($eventDay == "31") && ( ($eventMonth == "04")  || ($eventMonth == "06") || ($eventMonth ==
"09") || ($eventMonth == "11") ) ) {
      $date_error_31 = 1;
      $stage1_redirect = 1;
      //echo "bad date 31";
   }
   //Checking February issues
   if ( ($eventMonth == "02") && ( ($eventDay == "31") || ($eventDay == "30") ) ) {
      $date_error_feb = 1;
      $stage1_redirect = 1;
      //echo "bad feb date 1<br>";
   }

   if ( ($eventMonth == "02") && ($eventDay == "29") && (days_in_month(2, $eventYear)== 28) ) {
      $date_error_feb_leap = 1;
      $stage1_redirect = 1;
   }
?>
<html>
<head>
   <title>Bath Student Events</title>
   <?php
   redirect_to_login($redirect);
   if ($stage1_redirect == 1)
   {
   header("Location:../pages/newevent.php?error_format_event_name=".$error_format_event_name."&date_
error_31=".$date_error_31."&date_error_feb=".$date_error_feb."&date_error_feb_leap=".$date_error_feb
_leap."&PHPSESSID=".$PHPSESSID."");
      //echo "<META HTTP-EQUIV=\"refresh\"
CONTENT=\"5;URL=../pages/homepage.php?PHPSESSID=".$PHPSESSID."\">";
      //echo "<META HTTP-EQUIV=\"refresh\"
CONTENT=\"0;URL=../pages/newevent.php?error_format_event_name=".$error_format_event_name."&date_erro
r_31=".$date_error_31."&date_error_feb=".$date_error_feb."&date_error_feb_leap=".$date_error_feb_lea
p."\">";
   }
   ?>
   <LINK REL=StyleSheet HREF="http://www.bath.ac.uk/~ma0car/project/functions/styles.css"
TYPE="text/css">
</head>
<body>
   <?php
   //If we've got a date and it's valid, prepare it ready for posting to the setupplan.php
   if ($gotdate == 1) {
      $final_event_date = $eventYear."-".$eventMonth."-".$eventDay;
   //echo $final_event_date;
   } else {
      $final_event_date = "NULL";
   //echo $final_event_date;
   global $str_message;
   include '../functions/header.php';
   //echo "<br>";
   $str_message = "Set up a new event";
   include ("../functions/messagebar.php");
   //As the date is valid, set it up in a way we want to pass it.
   //This means we only post a date in the form below if we have one!
   ?>
   <table border="0" bordercolor = "black" cellpadding="2" cellspacing="0" width="656"
align="center">
      <tr>
         <td width="107" valign="top">
            <?php
               include 'http://www.bath.ac.uk/~ma0car/project/functions/toolbar.php';
            ?>
         </td>
         <td width ="549" valign="top">
            <table>
               <tr width ="549">
                  <td align = "left">
                     <?php
                        //Set up the relevant breadcrumbs
                        echo "<a href=\"../pages/homepage.php?PHPSESSID=".$PHPSESSID."\">";
                        echo "<font class = \"breadcrumb\"><strong>Home</font></a>";
                        echo " > ";
                        echo "<A HREF=\"../pages/newevent.php?PHPSESSID=".$PHPSESSID."\">";
                        echo "<font class = \"breadcrumb\"><strong>Create a new event: Stage
1</font></a>";
                        echo " > ";
                        //echo "<A HREF=\"../pages/wizard.php?PHPSESSID=".$PHPSESSID."\">";
                        echo "<font class = \"breadcrumb\"><strong>Stage 2</font>";
                     ?>
                  </td>
               </tr>
               <tr width ="540">
                  <td width ="540" height="300" valign="top">
                     <table border="2" cellpadding="2" cellspacing="2" style="border-collapse: collapse
overflow: scroll" bordercolor="#111111" width="100%" height="300" valign="top" bgcolor="FFFFCC">
                        <tr>
                           <td valign="top">
                              <font class = "big">Stage 2 - Event Wizard</font>
                              <br>
                              <font class = "small">By answering these 6 questions about your event, we
can set up a plan for you. If you are ensure of an answer, always over-estimate, or include it
anyway.
                              <br><br>
                              <form name="input" action="../functions/setupplan.php" method="post">
                                 <strong>1) How many people are you expecting at your event?</strong><br>
                                 Up to 30
                                 <input type="radio" name="numbers" value="1" checked
/>      
                                 31 - 100
                                 <input type="radio" name="numbers" value="2"
/>      
                                 Over 100
                                 <input type="radio" name="numbers" value="3"
/>      
                                 <br><br>
                                 <strong>2) Are you organising this event for a Students Union
society?</strong><br>
                                 Yes
                                 <input type="radio" name="susoc" value="1" checked
/>      
                                 No
                                 <input type="radio" name="susoc" value="2"
/>      
                                 <br><br>
                                 <strong>3) Might the event be held on campus?</strong><br>
                                 Yes
                                 <input type="radio" name="oncampus" value="1" checked
/>      
                                 No
                                 <input type="radio" name="oncampus" value="2"
/>      
                                 <br><br>
                                 <strong>4) If you are having food, where is it coming from?</strong>
                                  (Tick all that apply)<br>
                                 We are making our own food
                                 <input type="checkbox" name="ownfood" value="1" />
                                 <br />
                                 We are using Students Union / University Catering
                                 <input type="checkbox" name="intcat" value="1" />
                                 <br />
                                 We are using external Caterers (e.g. a local restaurant)
                                 <input type="checkbox" name="extcat" value="1" />
                                 <br />
                                 We are NOT having any food
                                 <input type="checkbox" name="nofood" value="1" />
                                 <br><br>
                                 <strong>5) Could any of the following describe your event?</strong>
 (Tick all that apply)<br>
                                 Pub Tour
                                 <input type="checkbox" name="pubtour" value="1" />
                                 <br />
                                 Theatre Trip
```

```
                        <input type="checkbox" name="theatre" value="1" />
                        <br />
                        Exhibition / Fayre
                        <input type="checkbox" name="fayre" value="1" />
                        <br />
                        Political Speaker
                        <input type="checkbox" name="political" value="1" />
                        <br><br>
                        <strong>6) Are you considering any of the following</strong>
                         (Tick all that apply)<br>
                        Having music or a band playing?
                        <input type="checkbox" name="music" value="1" />
                        <br />
                        Having a temporary structure? (e.g. Tent, marquee)
                        <input type="checkbox" name="tempstruct" value="1" />
                        <br />
                        Will the event be held in a public place?
                        <input type="checkbox" name="public" value="1" />
                        <br />
                        Will the event be held on or near a road?
                        <input type="checkbox" name="road" value="1" />
                        <br />
                        Having an internal bar?
                        <input type="checkbox" name="intbar" value="1" />
                        <br><br>
                        <?php
                        //Pass values from stage 1
                        echo "<input type=\"hidden\" name=\"eventname\"
value=\"".$eventname."\">";
                        echo "<input type=\"hidden\" name=\"final_event_date\"
value=\"".$final_event_date."\">";
                        //Pass the session ID
                        echo "<input type=\"hidden\" name=\"PHPSESSID\"
value=\"".$PHPSESSID."\">";
                        ?>
                        <input type="submit" value="Click here to set up your event plan"/>
                        </form>
                    </td>
                  </tr>
                </table>
              </td>
            </tr>
          </table>
        </td>
      </tr>
    </table>
</body>
</html>
```